

1. Introduction

The Tender Management System is a sophisticated web-based platform designed to revolutionize the process of managing tenders within organizations. Tenders play a crucial role in the procurement process, acting as invitations for suppliers to bid on contracts for the provision of goods or services. However, traditional tender management methods often suffer from inefficiencies, such as manual paperwork, cumbersome communication channels, and lack of transparency.

The Tender Management System addresses these challenges by offering a centralized, automated solution that streamlines every stage of the tendering process. From tender creation and publication to bid submission and evaluation, the system ensures transparency, efficiency, and accountability at every step.

1.1 Company profile / Institute profile / Client profile

CodTech IT Solutions: Empowering Business Growth

CodTech IT Solutions is a renowned Business and Management Consulting firm specializing in tailored solutions for businesses of all sizes. With a team of seasoned experts and a focus on excellence, we offer:

Comprehensive Consulting Services: From strategic planning to operational optimization, we provide a wide range of consulting services to address diverse business needs.

Cutting-Edge IT Solutions: Our expertise in Java backend development ensures robust and scalable IT solutions, including software development, system integration, and cybersecurity services. **Client-Centric Approach:** We understand that every business is unique, which is why we prioritize delivering personalized solutions that exceed client expectations.

1.2 Abstract

The Tender Management System is a pivotal innovation in the realm of procurement and contract management. Traditional methods of tender management often involve manual processes, leading to inefficiencies, delays, and potential for errors. This project presents a comprehensive solution to address these challenges through the development of a web-based platform.

The system automates the entire tendering process, from tender creation and publication to bid submission and evaluation. By centralizing data and streamlining workflows, it enhances transparency, efficiency, and accountability. Key features include user management, tender creation, bid submission, and evaluation tools, all accessible through an intuitive user interface.

This project documentation provides insights into the system's functionalities, requirements, architecture, design, implementation, and testing. The Tender Management System represents a significant advancement in modernizing procurement processes, paving the way for smoother operations and better outcomes for organizations worldwide.

1.3 Existing System and Need for system

Existing System:

In many organizations, tender management processes are often manual and paper-based, relying heavily on emails, phone calls, and physical documents. The existing systems suffer from several drawbacks:

- **Manual Processes:** Tender creation, publication, and evaluation involve a significant amount of manual effort, leading to inefficiencies and errors.
- **Limited Transparency:** Lack of centralized systems makes it difficult to track the status of tenders, resulting in reduced transparency and accountability.
- **Communication Challenges:** Coordinating communication between stakeholders, such as procurement teams, suppliers, and evaluators, can be cumbersome and time-consuming.
- **Data Security Risks:** Paper-based systems are prone to data loss, security breaches, and unauthorized access, posing risks to sensitive information.

Need of system:

The shortcomings of the existing tender management processes underscore the need for a comprehensive Tender Management System. The new system aims to address these challenges by offering the following benefits:

- **Automation:** By automating key processes such as tender creation, publication, and evaluation, the system reduces manual effort, minimizes errors, and improves efficiency.
- **Transparency:** With a centralized platform, stakeholders can easily track the status of tenders, access relevant documents, and monitor progress in real-time, enhancing transparency and accountability.
- **Enhanced Security:** By storing data in a secure, centralized database with access controls and encryption mechanisms, the system mitigates risks associated with data loss, breaches, and unauthorized access.

1.4 Scope of system

- **Tender Creation and Management:** The system allows authorized users to create, edit, and publish tender documents. Users can specify details such as deadlines, requirements, and evaluation criteria. It facilitates the categorization and classification of tenders based on industry, type, and value. Additionally, users can upload and manage tender-related documents.
- **Bid Submission:** Suppliers can register and submit bids for available tenders. They can access tender documents, prepare bids online, and track the status of their submissions. The system enables suppliers to receive notifications of updates or changes to tender requirements.
- **Tender Evaluation:** Authorized evaluators can access tender submissions, review documents, and evaluate bids based on predefined criteria. The system supports customizable scoring mechanisms and facilitates collaborative decision-making among evaluators. It ensures transparency and fairness in the tender evaluation process.
- **Administration and Security:** Administrators can manage user accounts, roles, and permissions to ensure proper access control and security. The system implements robust security measures, including encryption, authentication.
- **User Experience and Accessibility:** The system features a user-friendly interface with intuitive navigation and responsive design for seamless user experience across devices. It complies with accessibility standards and guidelines to ensure usability for users with disabilities. This ensures that the system is accessible and easy to use for all stakeholders involved in

1.5 Operating Environment

Hardware:

- Standard web server hardware
- Client Devices (PCs, laptops, tablets, smartphone)

Software:

- **Operating System:** Windows / Linux / MacOS
- **Web server:** Apache Tomcat
- **Database:** MySQL Server
- **Backend Language:** Java
- **Frontend Technologies:** HTML, CSS, JSP, Bootstrap
- **Ide:** Eclipse

1.6 Brief Description of technology used

- **HTML:** HTML (Hypertext Markup Language) is the standard markup language for creating web pages and web applications. It provides the structure and layout for web content, allowing developers to define the elements and attributes that make up a webpage.
- **CSS:** Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML. It enhances the appearance and layout of web pages, allowing developers to control aspects such as colors, fonts, spacing, and positioning.
- **JSP:** Java Server Pages (JSP) is a technology that enables the development of dynamic web pages based on Java technology. It allows developers to embed Java code into HTML pages, facilitating the creation of dynamic content and interactive web applications.
- **Java:** Java is a high-level, object-oriented programming language developed by Sun Microsystems (now owned by Oracle). It is widely used for building a variety of applications, from enterprise software to mobile apps and web applications.
- **JDBC:** Java Database Connectivity (JDBC) is an API (Application Programming Interface) that enables Java applications to interact with relational databases. It provides a set of Java classes and interfaces to establish database connections, execute SQL queries, and retrieve results.
- **Servlet:** A Servlet is a Java programming language class used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Servlets interact with clients via a request-response paradigm managed by a web container (e.g., Apache Tomcat, Jetty), enabling dynamic content generation on web servers.
- **Tailwind CSS:** Tailwind CSS is a highly customizable CSS framework that provides a utility-first approach to building modern web interfaces. It offers a comprehensive set of pre-built CSS utility classes, allowing developers to quickly style their HTML elements without the need to write custom CSS.
- **JavaScript:** JavaScript is a versatile programming language commonly used in web development to create interactive and dynamic elements on web pages. Initially developed for client-side scripting in web browsers, JavaScript has evolved to become a full-fledged programming language capable of both front-end and back-end development.
- **RDBMS used to build database (MySQL):** MySQL is utilized as the primary relational database management system (RDBMS) for building and managing the database of the Train Ticket Reservation System. MySQL offers a robust set of features for storing, retrieving, and manipulating structured data, making it well-suited for handling the data requirements of the system. Additionally, MySQL provides scalability, reliability, and performance optimizations.

2. Proposed System

2.1 Study of similar System

Studying similar systems can provide valuable insights into various aspects such as user experience, functionality, performance, and security features. Here's a suggested approach for conducting a study of similar systems for the Tender Management System project:

Identify Similar Systems: Start by identifying existing tender management systems or similar platforms. Look for both commercial solutions and open-source projects. Examples might include government procurement systems, commercial tender management software, or platforms used by organizations for procurement processes.

Review Features and Functionality: Explore the features and functionality offered by each system. Pay attention to core functionalities such as user authentication, tender creation and management, bidding processes, notification systems, reporting capabilities, and vendor management features. Document how each system handles these functionalities and any additional features they offer.

Evaluate User Experience: Assess the user experience (UX) of each system by analyzing their interface design, navigation structure, ease of use, and overall user satisfaction. Consider aspects such as intuitive user interfaces, accessibility features, and responsiveness across different devices.

2.2 Feasibility Study

A feasibility study is essential to assess the viability of implementing the Tender Management System within an organization. It involves evaluating various aspects such as technical, economic, operational, and scheduling feasibility. Here's an overview:

- **Technical Feasibility:** Hardware Requirements: Assess if the existing hardware infrastructure is capable of supporting the system or if upgrades are necessary.
- **Software Compatibility:** Determine if the required software components (e.g., web servers, databases) are compatible and available.
- **Scalability:** Evaluate if the system can scale to accommodate increasing volumes of tender management activities.
- **Economic Feasibility:** Cost-Benefit Analysis: Conduct a cost-benefit analysis to compare the costs of developing, implementing, and maintaining the system against the anticipated benefits.

- **Operational Feasibility:** User Acceptance: Assess if the system meets the requirements and expectations of end-users, including administrators, evaluators, and suppliers.
- **Data Privacy:** Ensure that the system complies with data privacy regulations and protects sensitive information, such as personal data and proprietary business information.
- **Risk Analysis:** Risks: Identify potential risks and challenges that may arise during the implementation and operation of the system.
- **Risk Mitigation:** Develop strategies to mitigate identified risks and minimize their impact on the project's success.
- **Resource Utilization:** Evaluate the environmental impact of the system in terms of energy consumption, resource utilization, and waste generation.
- **Sustainability:** Assess if the system supports sustainable practices and aligns with the organization's environmental objectives.

2.3 Objectives of the proposed system

The objectives of the proposed Tender Management System are as follows:

- **Streamline Tender Processes:** The system aims to automate and streamline the entire tendering process, from tender creation and publication to bid submission and evaluation. By reducing manual effort and paperwork, it enhances efficiency and accelerates the procurement cycle.
- **Enhance Transparency and Accountability:** The system facilitates transparency in tender management by providing stakeholders with real-time access to tender information, bid submissions, and evaluation results. This promotes accountability and fairness in the tendering process.
- **Optimize Resource Utilization:** By automating routine tasks and providing analytics tools for data-driven decision-making, the system helps organizations optimize resource utilization and make informed procurement decisions.
- **Ensure Security and Compliance:** The system implements robust security measures to protect sensitive data and ensure compliance with data privacy regulations. This safeguards confidential information and mitigates risks associated with data breaches.
- **Support Customization and Scalability:** The system offers customization options to adapt to the unique requirements and workflows of different organizations. Additionally, it is designed to scale efficiently to accommodate growing volumes of tender activities and users.

2.4 Users of system

1. Administrator:

- Responsible for overall system management.
- Manages tender creation, notice, and evaluation processes.
- Generates reports and monitors system performance.
- Provides technical support and resolves issues related to the system.

2. Vendor:

- Registers and maintains vendor profile information in the system.
- Receives notifications about available tenders.
- Accesses tender.
- Prepares and submits bids for tender opportunities.
- Tracks the status of submitted bids.

3. Analysis and Design

3.1 System Requirements

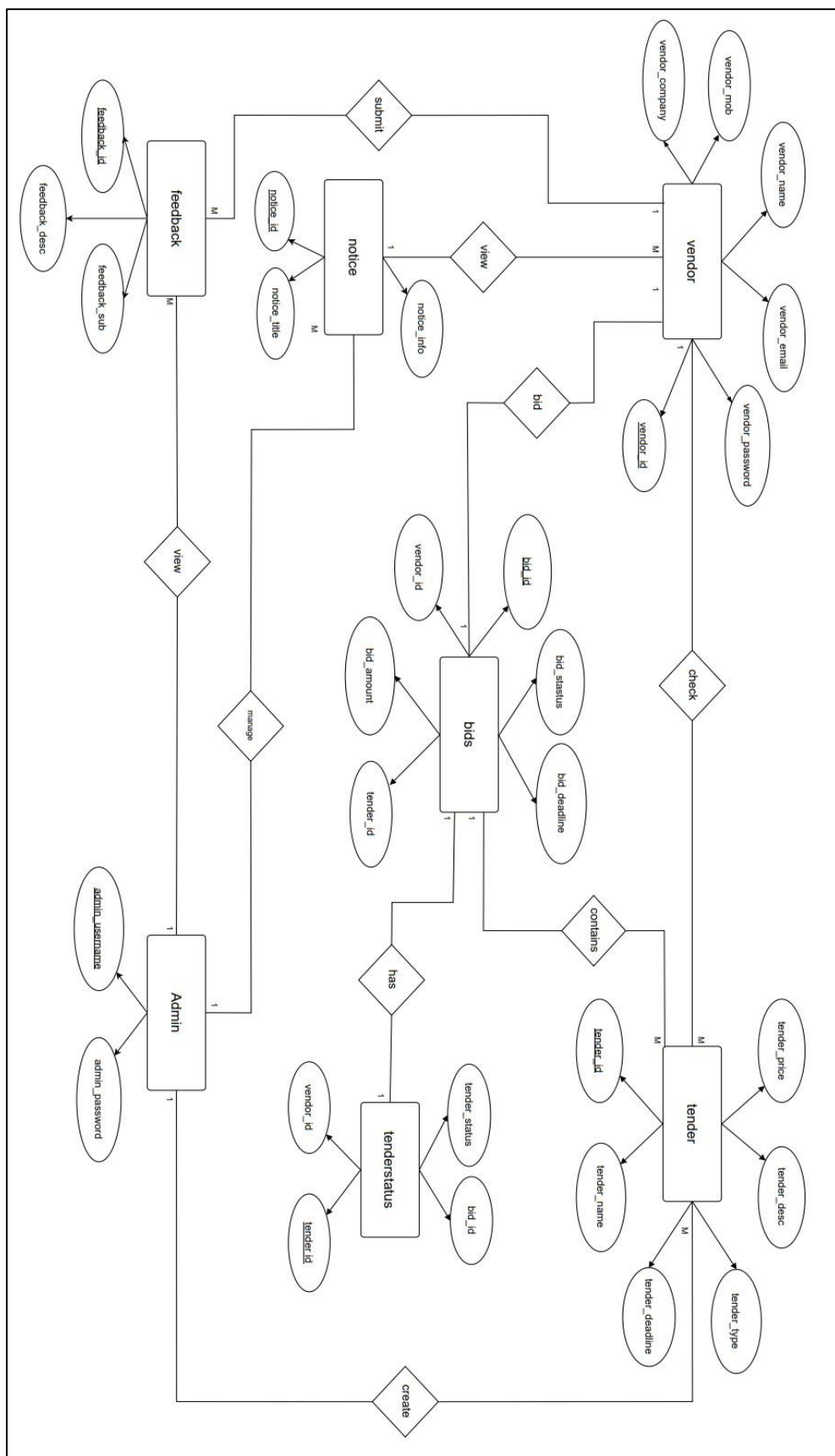
Functional requirement:

- User management system with roles for administrators, vendors, and bidders.
- Ability for administrators to create, edit, and publish tender documents with specified details.
- Platform for registered vendors and bidders to view, access, and submit bids for tender opportunities.
- Support for tender evaluation by authorized evaluators, including access to submissions and criteria review.
- Communication features for stakeholders and notifications for updates or changes to tender requirements.
- Reporting and analytics capabilities for monitoring tender activities and performance metrics.
- Implementation of robust security measures, role-based access control, and encryption for data protection.
- Integration with existing systems and customization options to adapt to organizational needs.

Non-Functional requirement:

- **Performance:** The system should respond promptly to user actions and handle concurrent users efficiently.
- **Usability:** The user interface should be intuitive, easy to navigate, and accessible to users with varying levels of technical expertise.
- **Reliability:** The system should be stable and dependable, with minimal downtime and robust error handling.
- **Scalability:** The system should accommodate increasing numbers of users and growing data volumes without significant degradation in performance.
- **Security:** The system should employ strong encryption, authentication, and authorization mechanisms to protect sensitive data and prevent unauthorized access.
- **Compatibility:** The system should be compatible with various web browsers and devices to ensure a consistent user experience.

3.2 Entity Relationship Diagram



3.3 Table Structure

- **Table Name: vendor**

This table is used to store details about the vendor details

Sr. No	Field	Data Type	Size	Constraint	Description
1.	vendor_id	int	11	PRIMARY KEY	Vendor Id
2.	vendor_name	varchar	20	NOT NULL	Vendor name
3.	vendor_email	varchar	20	NOT NULL	Vendor email
4.	vendor_password	varchar	20	NOT NULL	Vendor password
5.	vendor_mob	int	10	NOT NULL	Vendor mobile number
6.	vendor_address	varchar	50		Vendor address
7.	vendor_company	varchar	20	NOT NULL	Vendor company name

- **Table Name: tender**

This table is used to store details about the tender details

Sr. No	Field	Data Type	Size	Constraint	Description
1.	tender_id	int	11	PRIMARY KEY	Tender Id
2.	tender_name	varchar	20	NOT NULL	Tender name
3.	tender_type	varchar	20	NOT NULL	Tender Type
4.	tender_desc	varchar	30	NOT NULL	Tender description
5.	tender_price	int	10	NOT NULL	Tender price
6.	tender_loc	varchar	20		Tender lock
7.	tender_deadline	varchar	10	NOT NULL	Tender deadline date

- **Table Name: bid**

This table is used to store details about the bid details

Sr. No	Field	Data Type	Size	Constraint	Description
1.	bid_id	int	11	PRIMARY KEY	Bid id
2.	bid_status	varchar	20	NOT NULL	Bid status
3.	bid_deadline	varchar	20	NOT NULL	Bid status
4.	tender_id	int	11	FOREIGN KEY	Tender id
5.	bid_amount	int	10	NOT NULL	Bid amount
6.	vendor_id	int	11	FOREIGN KEY	Vendor id

- **Table Name: notice**

This table is used to store details about the notice details

Sr. No	Field	Data Type	Size	Constraint	Description
1.	notice_id	int	11	PRIMARY KEY	Notice id
2.	notice_title	varchar	20	NOT NULL	Notice title
3.	notice_sub	varchar	50	NOT NULL	Notice Subject

- **Table Name: tendorstatus**

This table is used to store details about the tendostatus details

Sr. No	Field	Data Type	Size	Constraint	Description
1.	tender_id	int	11	PRIMARY KEY	Tender id
2.	vendor_id	int	11	FOREIGN KEY	Vendor id
3.	bid_id	Int	11	FOREIGN KEY	Bid id
4.	tender_status	varchar	20	NOT NULL	Tender status

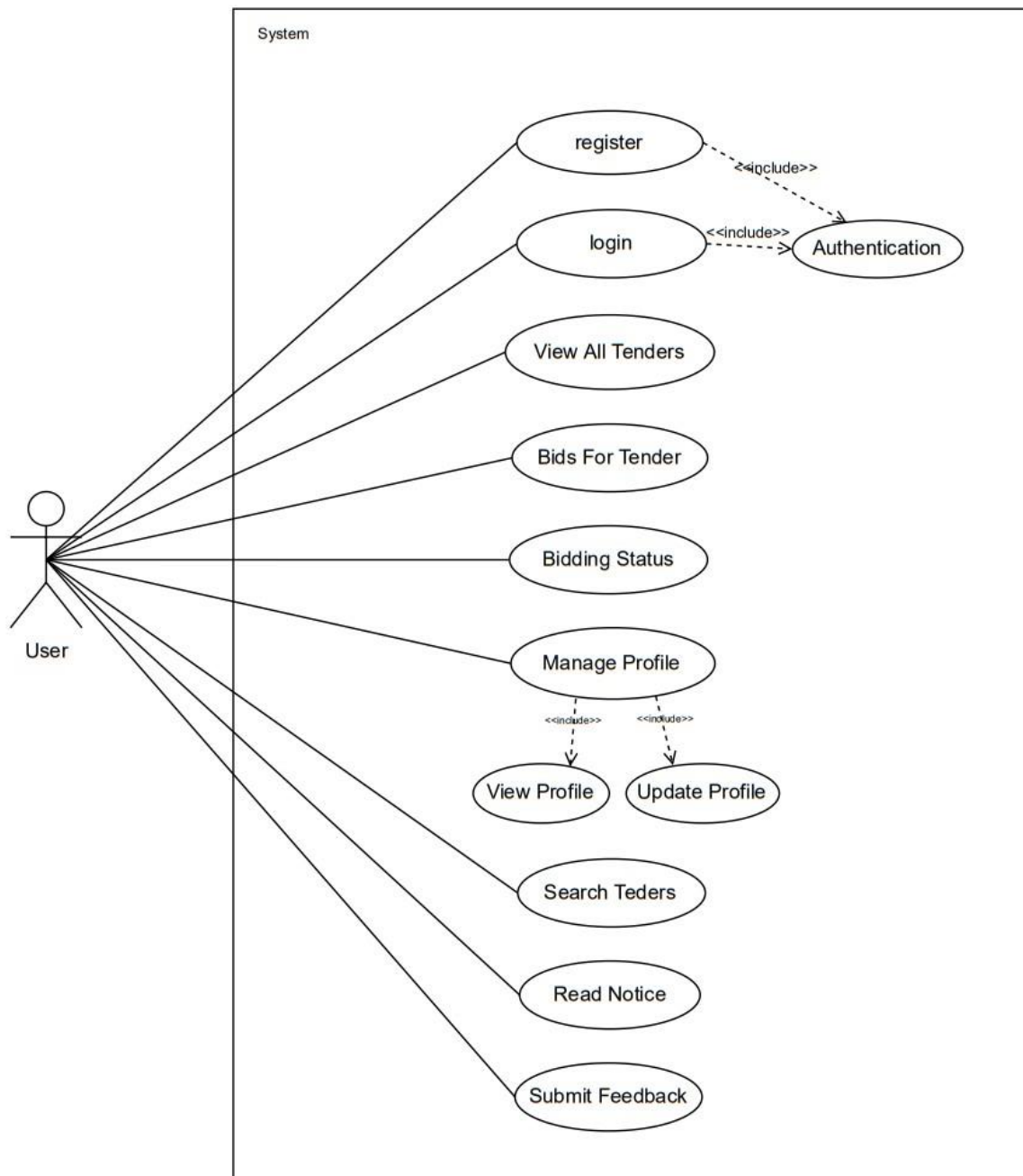
- **Table Name: feedback**

This table is used to store details about the feedback details

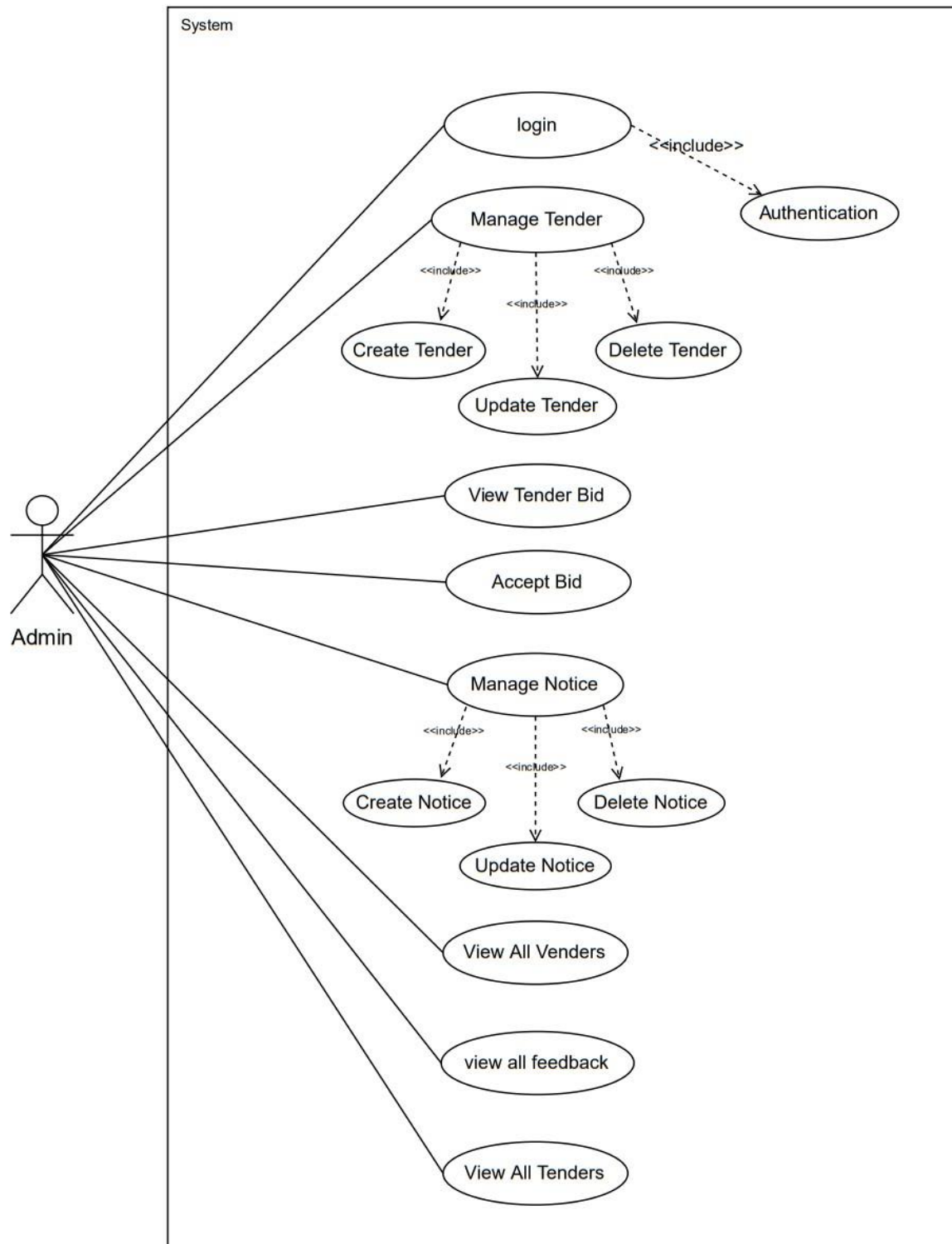
Sr. No	Field	Data Type	Size	Constraint	Description
1.	feedback_id	int	11	PRIMARY KEY	Feedback id
2.	feedback_title	varchar	20	NOT NULL	Feedback title
3.	feedback_desc	varchar	50	NOT NULL	Feedback Description
4.	vendor_id	int	11	FOREIGN KEY	Vendor Id
5.	vendor_email	varchar	20	NOT NULL	Vendor Email

3.4 Use case Diagram

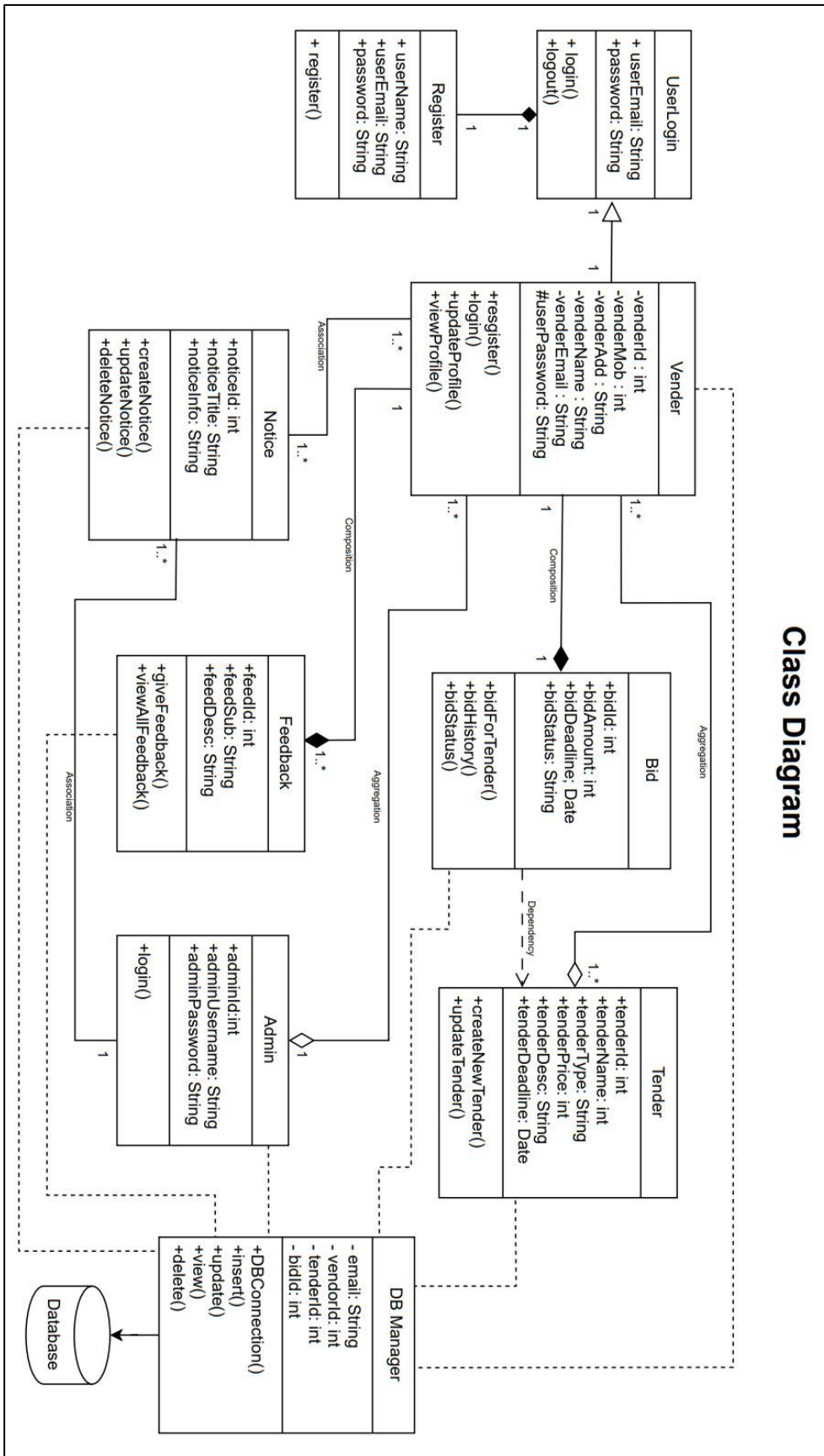
Use case diagram for User



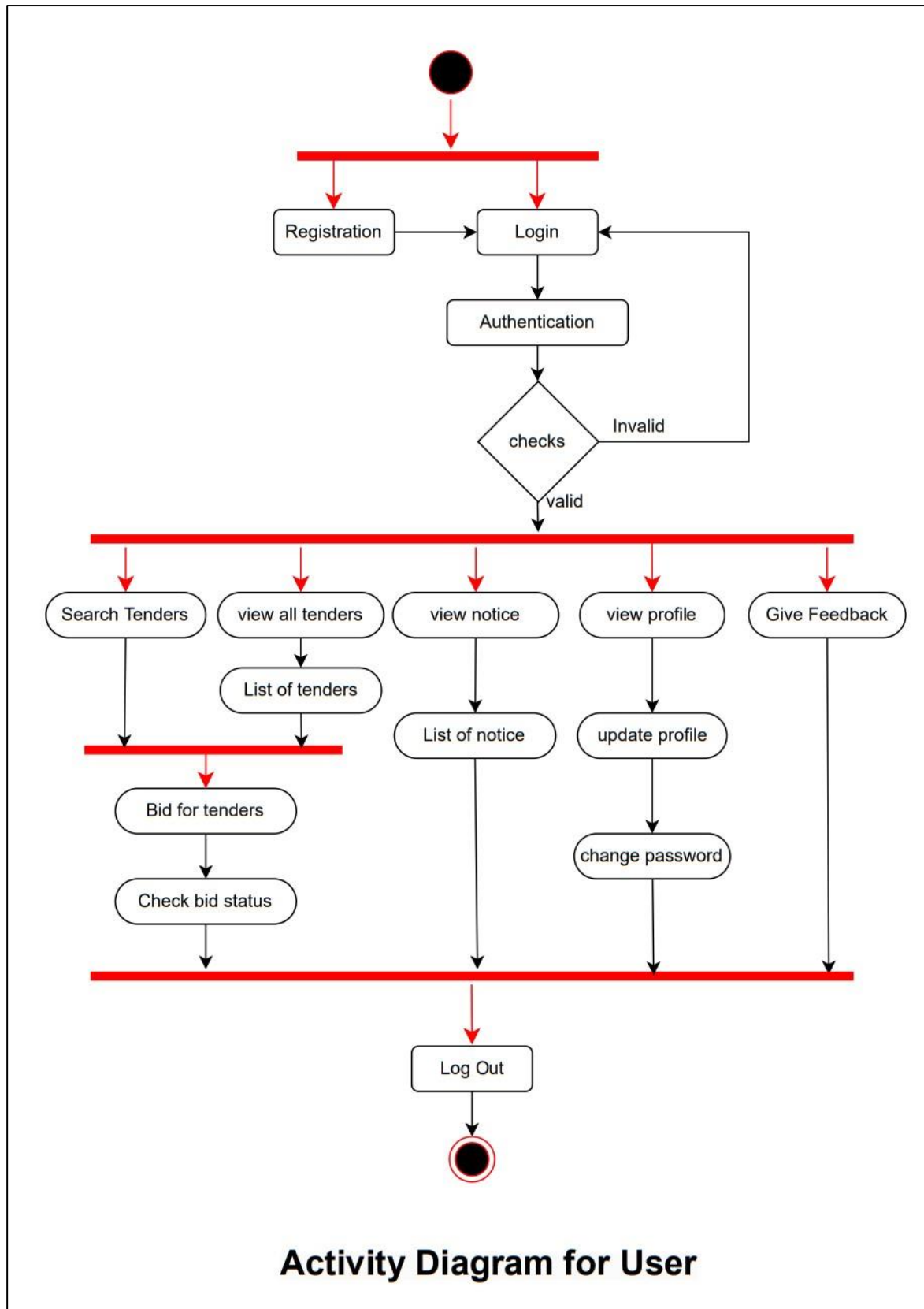
Use case diagram for Admin

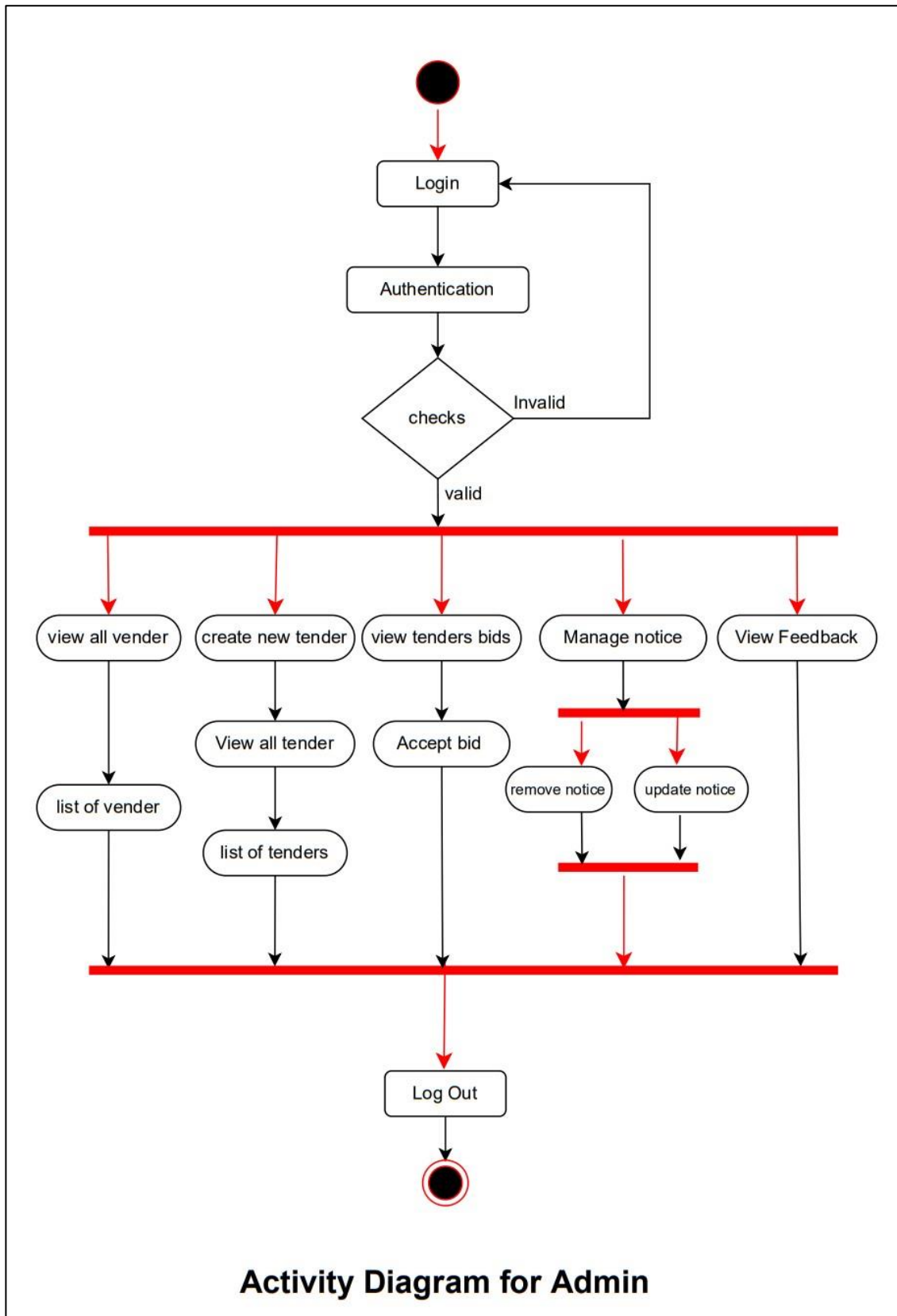


3.5 Class Diagram



3.6 Activity Diagram

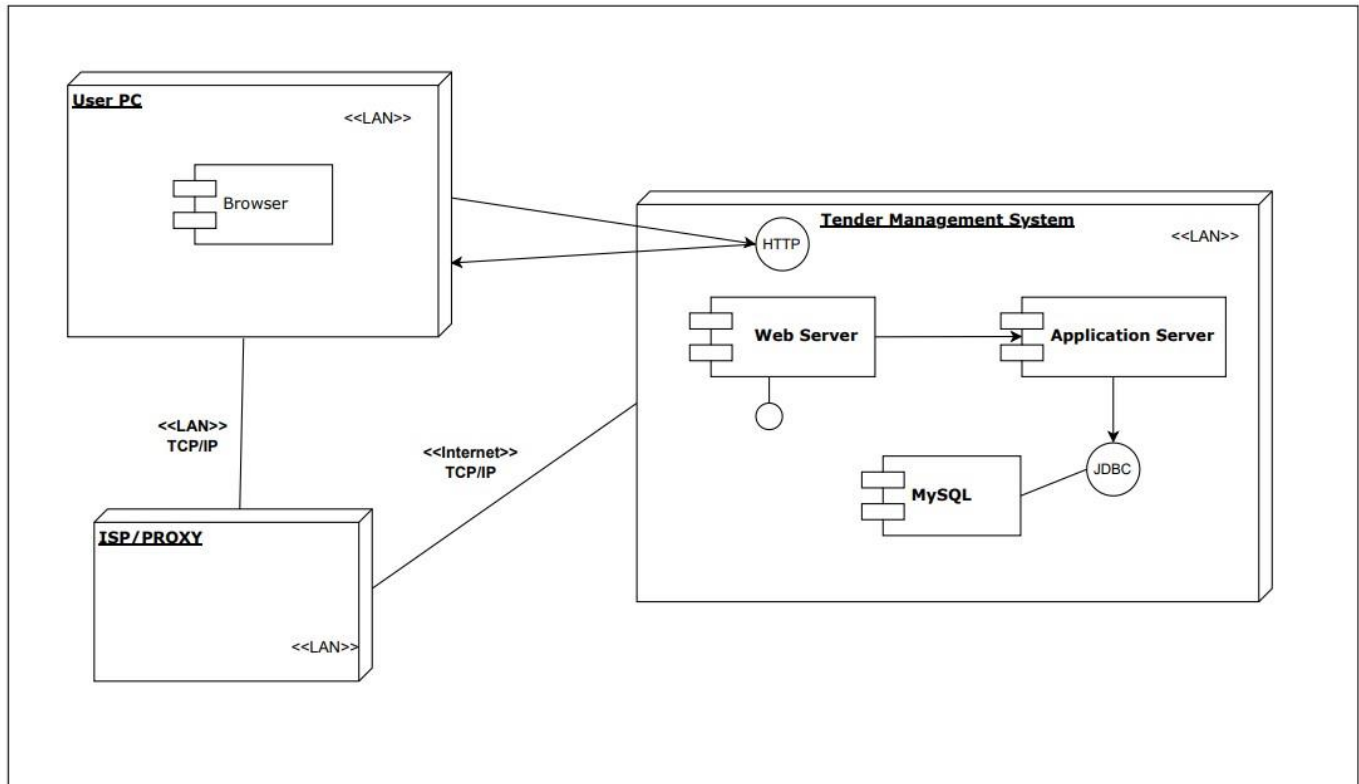




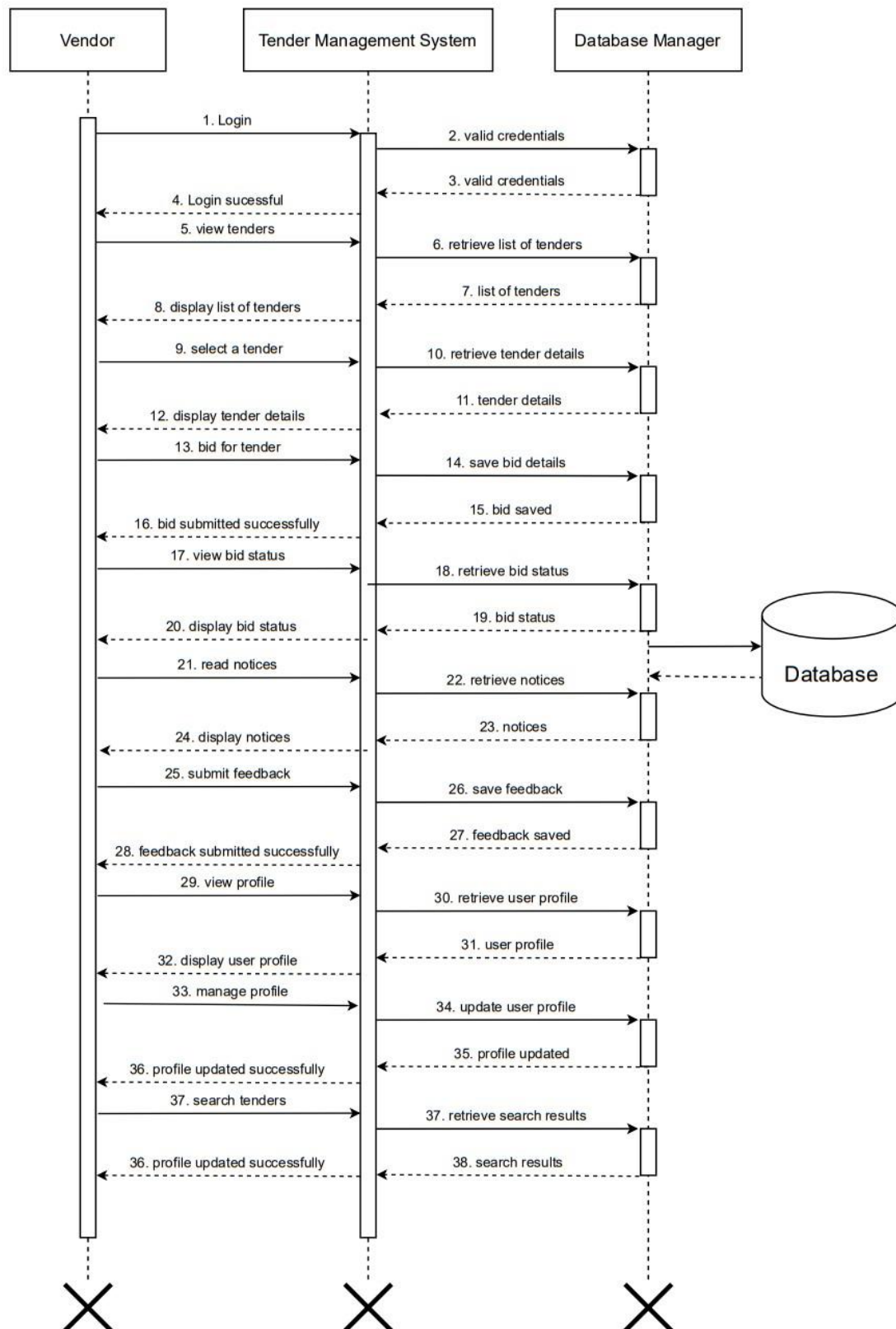
Activity Diagram for Admin

3.7 Deployment Diagram

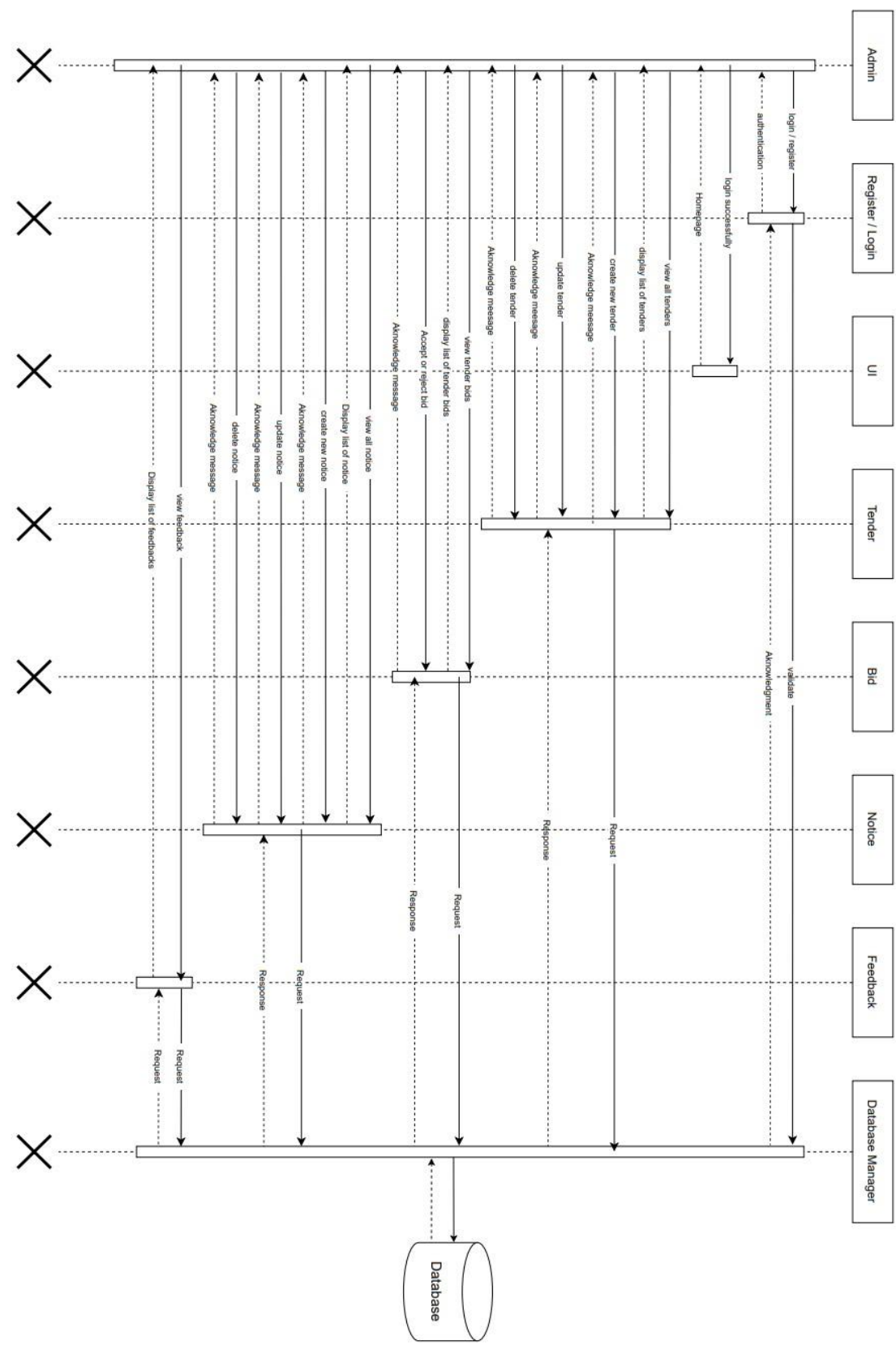
Deployment Diagram



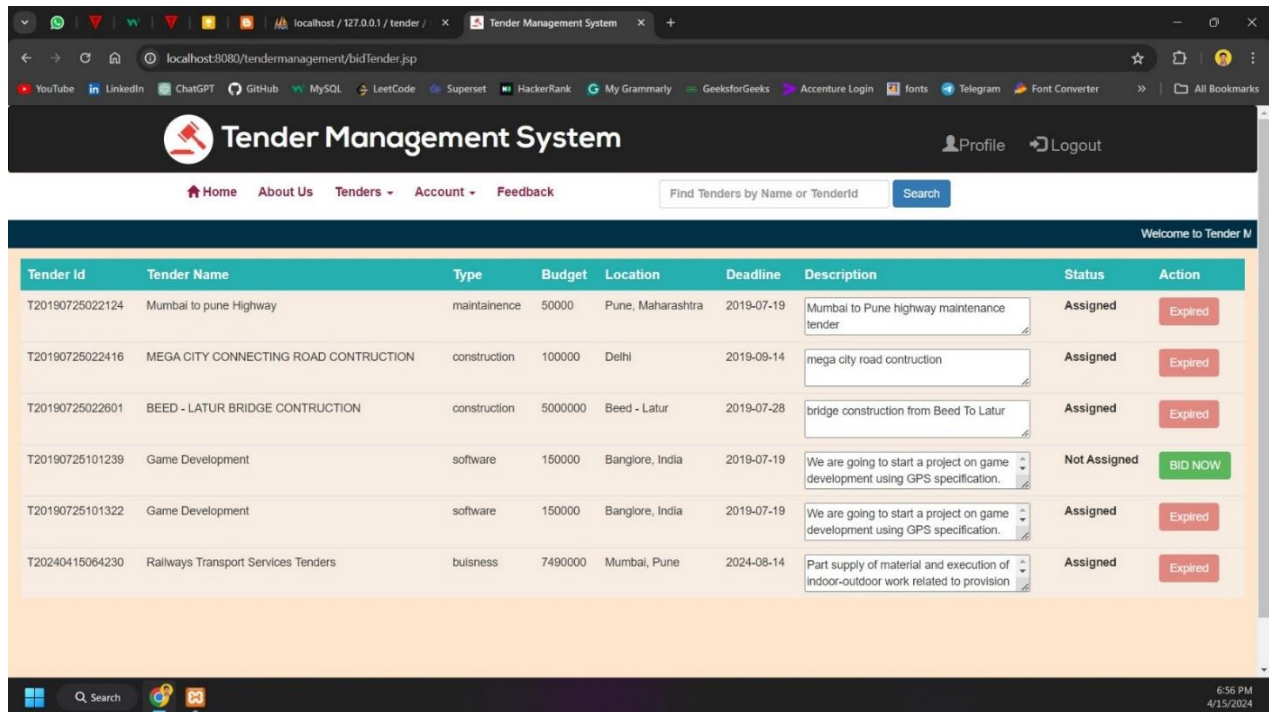
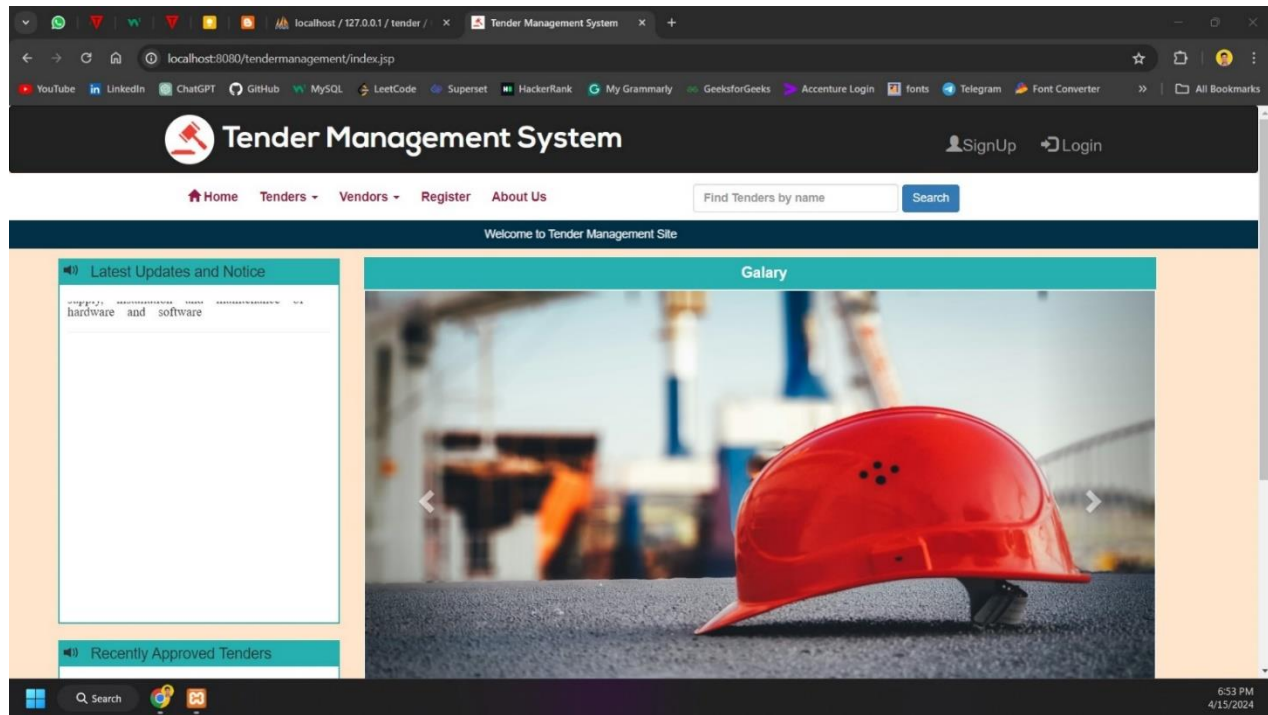
3.8 Sequence Diagram for Vender



3.9 Sequence Diagram for Admin



3.10 Sample of input screens



Latest Updates and Notice

BEED-LATUR BRIDGE CONSTRUCTION

ASSIGNED ENGINEER NEED TO REPORT AT THE CONSTRUCTION SITE BY TOMMOROW

Highway Reconstruction Tender

The Mumbai to Pune highway reconstruction tender is announced on 20 May

Metro Line extension

Recently Approved Tenders

Assigned To: T20190725022601

Application Id: B20190725101554

Assigned To: T20190725101322

Enter Feedback Below

Feedback

Id :	V20240405111809
Email :	suraj@gmail.com
Feedback Title :	<input type="text"/>
Feedback Descripton :	<input type="text"/>
<input type="button" value="Submit"/>	

Latest Updates and Notice

Metro Line extension

metro line extension tender is postponed to next year

Banking and mutual Funds

supply, installation and maintenance of hardware and software

BEED-LATUR BRIDGE CONSTRUCTION

ASSIGNED ENGINEER NEED TO REPORT AT THE CONSTRUCTION SITE BY TOMMOROW

Highway Reconstruction Tender

Recently Approved Tenders

Application Id: B20190725023512

Assigned To: T20190725022416

Application Id: B20190725024243

Edit Details And Click on Update

Update Profile

Vendor Id :	V20240405111809
Vendor Name:	<input type="text" value="Suraj Shinde"/>
Mobile No. :	<input type="text" value="8967453345"/>
Email Id :	<input type="text" value="suraj@gmail.com"/>
Address :	<input type="text" value="Dhanore, Pune, MH"/>
Company Name :	<input type="text" value="Softtech"/>
Verify Password:	<input type="text"/>
<input type="button" value="Update Profile"/>	

localhost / 127.0.0.1 / tender | x Tender Management System x +

localhost:8080/tendermanagement/adminViewVendor.jsp

YouTube LinkedIn ChatGPT GitHub MySQL LeetCode Superset HackerRank My Grammarly GeeksforGeeks Accenture Login fonts Telegram Font Converter All Bookmarks

Tender Management System

Profile Logout

Home Vendors Tender Notice Feedback Logout

Find Tenders by name or ten Search

Welcome to Tender M

Latest Updates and Notice

Metro Line extension

metro line extension tender is postponed to next year

Busline and mutual Funds

Recently Approved Tenders

Application Id: B20190725101554
Assigned To: T20190725101322

Application Id: B20190725023010
Assigned To: T20190725022124

Application Id: B20240415064650

These are the Vendors Currently Registered With Us

Vendor Id	Vendor Name	Mobile	Email	Company	Address
V20190725020951	sanket pokharkar	9022785621	sanket10@gmail.com	GitSolution	Chakrapani vasahat, Bhosari, Pune 411039
V20190725022813	yogesh pawar	12345679	yog@gmail.com	sthatpatya solut	Dighi, Pune, Maharashtra
V20190725023446	Mayank Sharma	9851273081	mayank@gmail.com	Infosys	Lohegaon, Pune, MH
V20240405111809	Suraj Shinde	8967453345	suraj@gmail.com	Softtech	Dhanore, Pune, MH

7:03 PM 4/15/2024

localhost / 127.0.0.1 / tender | x Tender Management System x +

localhost:8080/tendermanagement/viewfeedback.jsp

YouTube LinkedIn ChatGPT GitHub MySQL LeetCode Superset HackerRank My Grammarly GeeksforGeeks Accenture Login fonts Telegram Font Converter All Bookmarks

Feedback Id : 1

Vendor Id : V20240405111809

Vendor Email : suraj@gmail.com

Feedback Title : Lorem ipsum dolor sit amet.

Feedback Description : Lorem, ipsum dolor sit amet consectetur adipisicing elit. Illum

Feedback Id : 3

Vendor Id : V20190725100730

Vendor Email : shashi@gmail.com

Feedback Title : Lorem ipsum dolor sit amet.

Feedback Description : Lorem, ipsum dolor sit amet consectetur adipisicing elit. Illum

7:14 PM 4/15/2024

4. Coding

4.1 Algorithms

User Authentication Algorithm:

1. Receive Request:
 - Receive a POST request containing the user's credentials (username and password).
2. Authenticate User:
 - Verify the user's credentials against the database records.
 - Check if the user exists and the password matches the stored hash.
3. Handle Authentication Result:
 - If authentication is successful:
 - Redirect the user to their respective dashboard page (Admin Home, Vendor Home, or Bidder Home).
 - Display a welcome message and provide access to relevant functionalities.
 - If authentication fails:
 - Redirect the user back to the login page.
 - Display an error message indicating invalid credentials

Tender Creation Algorithm:

1. Receive Request:
 - Receive a POST request containing tender details (title, description, deadlines, etc.).
2. Validate Inputs:
 - Ensure all required fields are filled out and deadlines are in the future.
3. Create Tender:
 - Insert the tender details into the database as a new record.
 - Generate a unique identifier for the tender.
4. Handle Creation Result:
 - If successful, redirect the user to the Tender List page.
 - If unsuccessful, display an error message and prompt the user to retry.

Bid Submission Algorithm:

1. Receive Request:
 - Receive a POST request containing bid details (bid amount, proposal, etc.).
2. Validate Bid:
 - Check if the bid amount is valid and within budget constraints.
3. Submit Bid:
 - Insert the bid details into the database associated with the respective tender.
4. Handle Submission Result:
 - If successful, redirect the user to the Bid History page.
 - If unsuccessful, display an error message and prompt the user to retry.

Notice Creation Algorithm:

1. Receive Request:
 - Receive a POST request containing notice details (title, content, visibility, etc.).
2. Validate Inputs:
 - Ensure all required fields are filled out.
3. Create Notice:
 - Insert the notice details into the database as a new record.
 - Generate a unique identifier for the notice.
4. Handle Creation Result:
 - If successful, redirect the user to the Notice List page.
 - If unsuccessful, display an error message and prompt the user to retry.

Feedback Submission Algorithm:

1. Receive Request:
 - Receive a POST request containing feedback details (user ID, feedback content, etc.).
2. Validate Inputs:
 - Ensure all required fields are filled out.
3. Submit Feedback:
 - Insert the feedback details into the database as a new record.
4. Handle Submission Result:
 - If successful, display a confirmation message to the user.
 - If unsuccessful, display an error message and prompt the user to retry.

4.2 Code Snippets

❖ AdminHome.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@page import="java.sql.*, com.hit.utility.DBUtil,
    javax.servlet.annotation.WebServlet, com.hit.beans_VENDORBean "
    errorPage="errorpage.jsp"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
    <head>
        <link rel="shortcut icon" type="image/png" href="images/Banner_Hit.png">
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Tender Management System</title>
        <link rel="stylesheet" href="css/bootstrap.min.css">
        <link rel="stylesheet" href="css/animate.css">
        <link href="css/font-awesome.min.css" type="text/css" rel="stylesheet">
        <link href="css/SpryTabbedPanels.css" type="text/css" rel="stylesheet">
        <link href="https://fonts.googleapis.com/css?family=Black+Ops+One"
rel="stylesheet">
        <link href="css/bootstrap-dropdownhover.min.css">
        <link rel="stylesheet" href="css/style2.css">
    </head>
    <body>

        <%
            String user = (String)session.getAttribute("user");
            String uname = (String)session.getAttribute("username");
            String pword = (String)session.getAttribute("password");

            if(!user.equalsIgnoreCase("user") || uname.equals("") ||
pword.equals("")){

                response.sendRedirect("loginFailed.jsp");

            }

        %>
```

```

<jsp:include page="header.jsp"></jsp:include>

<jsp:include page="vendorMenu.jsp"></jsp:include>

<jsp:include page="slider.jsp"></jsp:include>

<div class="container-fluid">

    <div class="notice">
        <div class="col-md-3" style="margin-left:2%">
            <% Connection con = DBUtil.provideConnection(); %>

            <jsp:include page="notice.jsp"></jsp:include><br>

            <jsp:include page="approved.jsp"></jsp:include><br>

        </div>
    </div>
    <%

        VendorBean vendor = (VendorBean)session.getAttribute("vendordata");

    %>

    <div class="col-md-8">
        <div class="marquee" style="border:2px black hidden; background-
color:white">
            <h4 style="background-color:#26b0b0; margin-top:-1.8px; margin-
bottom:1px;padding: 15px; text-align: center;color:black;font-weight:bold">
                &nbsp;   <span id="pagetitle">VENDOR ACCOUNT</span></h4>
            <div class="marquee-content" style="align:center; padding-
top:200px;min-height:750px;background-color:white">
                <h1><center>Hey <%= vendor.getName()%>! Welcome to Our Tender
Management system</center></h1>
                <h2><center>Here You can manage your tenders,view tenders and bid
for tenders according to their deadline and base price</center></h2>

            </div>
        </div>
    </div>
    <a><h1></h1></a>

</div>
<jsp:include page="footer.jsp"></jsp:include>
</body>

```

```
</html>
```

❖ AdminHome.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@page import="java.sql.*, com.hit.utility.DBUtil,
    javax.servlet.annotation.WebServlet" errorPage="errorpage.jsp"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html lang="en">
    <head>
        <link rel="shortcut icon" type="image/png" href="images/Banner_Hit.png">
        <!--link rel="shortcut icon" type="image/ico" href="images/hit_favicon.ico"-->

        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Tender Management System</title>
        <link rel="stylesheet" href="css/bootstrap.min.css">
        <link rel="stylesheet" href="css/animate.css">
        <link href="css/font-awesome.min.css" type="text/css" rel="stylesheet">
        <link href="css/SpryTabbedPanels.css" type="text/css" rel="stylesheet">
        <!--link rel="stylesheet" href="css/styles.css"-->
        <link href="https://fonts.googleapis.com/css?family=Black+Ops+One"
rel="stylesheet">
        <link href="css/bootstrap-dropdownhover.min.css">
        <link rel="stylesheet" href="css/style2.css">
    </head>
    <body>

        <%
            String user = (String)session.getAttribute("user");
            String uname = (String)session.getAttribute("username");
            String pword = (String)session.getAttribute("password");

            if(!user.equalsIgnoreCase("admin") || uname.equals("") ||
pword.equals("")){

                response.sendRedirect("loginFailed.jsp");

            }

        %>
        <jsp:include page="header.jsp"></jsp:include>
```

```

<jsp:include page="adminMenu.jsp"></jsp:include>

<jsp:include page="slider.jsp"></jsp:include>

<div class="container-fluid">

    <div class="notice">
        <div class="col-md-3" style="margin-left:2%">
            <% Connection con = DBUtil.provideConnection(); %>

            <jsp:include page="notice.jsp"></jsp:include><br>

            <jsp:include page="approved.jsp"></jsp:include><br>

        </div>
    </div>

    <div class="col-md-8">
        <div class="marquee" style="border:2px black hidden; background-color:white">
            <h4 style="background-color:#26b0b0; margin-top:-1.8px; margin-
bottom:1px;padding: 15px; text-align: center;color:white;font-weight:bold">
                &nbsp;    <span id="pagetitle">Admin Account</span></h4>
            <div class="marquee-content" style="align:center; padding-top:200px;min-
height:750px;background-color:white">
                <h1><center>Hey Admin! Welcome to Our Tender Management
system</center></h1>
                <h2><center>Here You can manage your vendors, add tenders and assign
it to vendors according to their high bids</center></h2>
            </div>
        </div>
    </div>

    <a><h1></h1></a>

</div>
<jsp:include page="footer.jsp"></jsp:include>
</body>
</html>

```

❖ VendorBean.Java

```
package com.hit.beans;

import java.io.Serializable;
import java.text.SimpleDateFormat;
import java.util.Date;

public class VendorBean implements Serializable{
    private String id;
    private String name;
    private String mobile;
    private String email;
    private String address;
    private String company;
    private String password;
    public VendorBean(){
    }
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getMobile() {
        return mobile;
    }
    public void setMobile(String mobile) {
        this.mobile = mobile;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
```

```

        this.address = address;
    }
    public String getCompany() {
        return company;
    }
    public void setCompany(String company) {
        this.company = company;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
    public VendorBean(String id,String name, String mobile, String email, String
address,
        String company, String password) {
        super();
        this.id = id;
        this.name = name;
        this.mobile = mobile;
        this.email = email;
        this.address = address;
        this.company = company;
        this.password = password;
    }
}

```

❖ VendorDao.Java

```
package com.hit.dao;

import java.util.List;

import com.hit.beans_VENDORBean;

public interface VendorDao {

    public String registerVendor(VendorBean vendor);

    public List<VendorBean> getAllVendors();

    public boolean validatePassword(String vendorId,String password);

    public String updateProfile(VendorBean vendor);

    public String changePassword(String vendorId,String oldPassword,String
newPassword);

    public VendorBean getVendorDataById(String vendorId);
}
```


5. Testing

5.1 Test Strategy

#Step 1: Test Scope and Objectives:

- Define the scope of testing, including the specific functionalities, modules, and components that need to be tested.
- Identify the objectives of testing, such as ensuring the system's functionality, usability, performance, security, and compatibility.

#Step 2: Test Approach:

- Determine the testing approach, such as whether to use manual testing, automated testing, or a combination of both.
- Specify the tools and frameworks to be used for test automation, if applicable.

#Step 3: Test Environment:

- Define the test environment requirements, including the hardware, software, and network configurations needed to perform the tests effectively.
- Ensure that the test environment closely resembles the production environment to obtain realistic results.

#Step 4: Test Levels:

- Determine the different levels of testing to be performed, such as unit testing, integration testing, system testing, and acceptance testing.
- Specify the responsibilities and roles of the testing team members for each level of testing.

#Step 5: Test Schedule:

- Create a timeline for the testing activities, taking into account the project's overall schedule and milestones.
- Define the test cycles and iterations, allocating time for defect reporting, retesting, and regression testing.

#Step 6: Risks and Mitigation:

- Identify potential risks and challenges associated with testing, such as resource constraints, time limitations, and dependencies on external systems.
- Develop a plan to mitigate or minimize these risks, including contingency measures if required.

#Step 7: Test Team and Communication:

- Specify the roles and responsibilities of the testing team members, including testers, test leads, and stakeholders.
- Establish effective communication channels and reporting mechanisms to facilitate collaboration and transparency.

#Step 8: Exit Criteria:

- Define the criteria that must be met to consider the testing phase complete and ready for the next stage or production deployment.
- This may include achieving a certain level of test coverage, resolving critical defects, and obtaining approval from stakeholders

5.2 Unit Test Plan

Unit Test Plan: Tender Management System Web Project

1. Introduction

The Unit Test Plan outlines the strategy and approach for testing individual units or components of the Tender Management System. This plan focuses on verifying that each unit performs as expected and meets its design specifications.

2. Objectives

- To verify the functionality and behavior of each unit in isolation.
- To ensure that each unit meets its requirements and design specifications.
- To identify and fix defects in individual units early in the development process.

3. Scope

The Unit Test Plan covers the testing of all individual units or modules of the Tender Management System, including:

- Vendor management
- Tender creation and management

- Bid submission
- Tender evaluation
- Notice evaluation
- Feedback Evaluation

4. Test Environment

- Programming Language: Java
- Testing Framework: JUnit or equivalent
- Integrated Development Environment (IDE): Eclipse, IntelliJ IDEA, etc.
- Web Browser: Google Chrome, Mozilla Firefox, etc.

5. Test Cases

- Each unit will have a set of test cases covering various scenarios to ensure thorough testing. Test cases will include:
- Positive test cases to verify that units perform as expected under normal conditions.
- Negative test cases to validate error handling and boundary conditions.
- Edge cases to test extreme or unexpected scenarios.

6. Test Execution

- Developers will write unit tests for each unit/module before implementing the code.
- Test cases will be executed using automated testing frameworks.
- Test results will be reviewed by developers to identify and fix any defects.
- Regression testing will be performed to ensure that changes do not introduce new defects.

7. Test Reporting

- Test results will be documented, including pass/fail status for each test case.
- Defects identified during testing will be logged in a defect tracking system.
- Test reports will be shared with the development team for review and resolution of issues.

8. Responsibilities

- Developers are responsible for writing unit tests for their code and ensuring test coverage.
- Quality Assurance (QA) team members will review test results and provide feedback to developers.
- Project managers will oversee the testing process and ensure that testing objectives are met.

9. Risks and Mitigation

- Risk: Incomplete test coverage
 - Mitigation: Conduct code reviews to identify areas with low test coverage and prioritize additional testing.
- Risk: Changes in requirements
 - Mitigation: Maintain close communication with stakeholders to address changes promptly and update test cases accordingly.

10. Conclusion

The Unit Test Plan outlines the approach for testing individual units of the Tender Management System to ensure that each component functions correctly and meets its requirements. By following this plan, we aim to deliver a high-quality system that meets user expectations and supports efficient tender management processes.

5.3 Acceptance Test Plan

1. Introduction

The Acceptance Test Plan outlines the strategy and approach for testing the Tender Management System from end to end to ensure that it meets the requirements and expectations of stakeholders. Acceptance testing verifies that the system behaves as intended and fulfills its intended purpose.

2. Objectives

- To verify that the Tender Management System meets all specified requirements and functional specifications.
- To ensure that the system is user-friendly, intuitive, and meets the needs of stakeholders.
- To validate that the system functions correctly in a production-like environment.

3. Scope

The Acceptance Test Plan covers testing of the entire Tender Management System, including all user roles, features, and functionalities. This includes:

- User registration and authentication
- Tender creation and management
- Bid submission and evaluation

- Reporting and analytics
- Security features
- Integration and customization components

4. Test Environment

- Web Browser: Google Chrome, Mozilla Firefox, etc.
- Test Data: Sample tender documents, user accounts, and test scenarios.
- Production-like environment: A server environment that closely resembles the production environment.

5. Test Cases

- Test cases will cover various scenarios to validate the system's functionality and behavior. Test cases will include:
- Positive test cases to verify that the system behaves as expected under normal conditions.
- Negative test cases to test error handling and boundary conditions.
- End-to-end test cases to validate complete user workflows, from tender creation to bid evaluation.

6. Test Execution

- Testers will execute test cases in the acceptance testing environment, following predefined test scripts.
- Testers will record test results, including pass/fail status and any defects encountered.
- Any issues identified during testing will be logged in a defect tracking system for resolution.

7. User Acceptance Testing (UAT)

- End users, including administrators, vendors, and bidders, will participate in User Acceptance Testing.
- Users will perform typical tasks and workflows within the system to validate its usability and effectiveness.
- Feedback from users will be collected and incorporated into the system as needed.

8. Test Reporting

- Test results will be documented, including pass/fail status for each test case.
- Defects identified during testing will be logged in a defect tracking system, along with severity and priority.

- Test reports will be shared with stakeholders, including project managers, developers, and end users.

9. Responsibilities

- Testers are responsible for executing test cases and reporting test results.
- Developers are responsible for addressing and resolving any defects identified during testing.
- Project managers will oversee the testing process and ensure that testing objectives are met.

10. Conclusion

The Acceptance Test Plan outlines the approach for validating the Tender Management System to ensure that it meets the requirements and expectations of stakeholders. By following this plan, we aim to deliver a high-quality system that enables efficient and transparent tender management processes.

5.3 Test Case / Test Script

1. Vendor Registration and Login

Test Case ID	Test Case Name	Test Case Description	Test Steps	Expected Result	Actual Result	Test Status (P/F)
VM_TC_001	Vendor Registration	Verify successful vendor registration.	1. Navigate to the vendor registration page	User sees the registration form.	Registration Form Display	P
			2. Fill in the required details. Click on the "Register" button.	The vendor should be registered successfully and redirected to the login page.	Vendor registered successfully and redirected to the login page.	P
VM_TC_002	Vendor Login	Verify successful login for registered vendors.	1. Enter valid login credentials. Click on the "Login" button.	The vendor should be logged in successfully and redirected to the vendor dashboard.	Vendor logged in successfully and redirected to the vendor dashboard.	P
VM_TC_003	Invalid Registration	Verify error message for invalid vendor registration.	1. Navigate to the vendor registration page.	User sees the registration form	Registration Form Display	P
			2. Enter invalid registration details. Click on the "Register" button.	An error message should be displayed indicating invalid registration details.	Error message displayed for invalid registration details.	P
VM_TC_004	Invalid Login	Verify error message for invalid vendor login.	1. Enter invalid login credentials. Click on the "Login" button.	An error message should be displayed indicating invalid login credentials.	Error message displayed for invalid login credentials.	P

2. Tender Module

Test Case ID	Test Case Name	Test Case Description	Test Steps	Expected Result	Actual Result	Test Status (P/F)
TM_TC_001	Create Tender	Verify successful creation of a new tender.	1. Navigate to the tender creation page.	Tender registration page should be display	Tender registration page is display	P
			2. Fill in the required details for the new tender. Click on the "Create Tender" button.	The new tender should be created successfully and listed in the tender dashboard	New tender created successfully and listed in the tender dashboard.	P
TM_TC_002	Edit Tender	Verify the ability to edit an existing tender.	1. Navigate to the list of created tenders.	List of all tender should be visible	List of all tender is display	P
			2. Select a tender to edit. Modify the tender details. Click on the "Save Changes" button.	The changes to the tender should be saved successfully.	Changes to the tender saved successfully.	P
TM_TC_003	Tender Search	Verify the functionality to search for specific tenders.	1. Navigate to the tender search page. Enter search criteria Click on the "Search" button.	The system should display a list of tenders matching the search criteria.	System displayed a list of tenders matching the search criteria.	P
TM_TC_004	Tender Deletion	Verify the ability to delete a tender.	1. Navigate to the list of created tenders.	List of all tender should be visible	List of all tender is display	P
			2. Select a tender to delete. Click on the "Delete" button	The selected tender should be deleted successfully from the system.	Selected tender deleted successfully from the system.	P

3. Notice Module

Test Case ID	Test Case Name	Test Case Description	Test Steps	Expected Result	Actual Result	Test Status (P/F)
NM_TC_001	Create Notice	Verify successful creation of a new notice.	1. Navigate to the notice creation page.	Notice page should be display	Notice page is display	P
			2. Fill in the required details for the new notice. Click on the "Create Notice" button.	The new notice should be created successfully and displayed in the notice board.	New notice created successfully and displayed in the notice board.	P
NM_TC_002	Update Notice	Verify successful updating of a notice.	1. Navigate to the notice creation page.	Notice page should be display	Notice page is display	P
			2. Fill in the required details for the update notice. Click on the "Update Notice" button.	The notice should be updated successfully and displayed in the notice board.	New notice updated successfully and displayed in the notice board.	P
NM_TC_003	Delete Notice	Verify successful deleting of a notice.	1. Navigate to the notice creation page.	Notice page should be display	Notice page is display	P
			2. Select the notice which want to deleted. Click on the "Delete Notice" button.	The notice should be deleted successfully and remove from the notice board.	New notice deletes successfully and remove from the notice board.	P

4. Bids Module

Test Case ID	Test Case Name	Test Case Description	Test Steps	Expected Result	Actual Result	Test Status (P/F)
BM_TC_001	Submit Bid	Verify successful submission of a bid for a tender.	1. Navigate to the list of available tenders.	List of all tender should be visible	List of all tender is display	P
			2. Select a tender to bid on. Fill in bid details. Click on the "Submit Bid" button.	The bid should be submitted successfully, and the vendor should receive confirmation.	Bid submitted successfully, and confirmation received.	P
BM_TC_002	Edit Bid	Verify the ability to edit a submitted bid.	1. Navigate to the list of submitted bids.	List of all bids should be visible	List of all bids is display	P
			2. Select a bid to edit. Click on the "Edit" button.	The bid details should be displayed for editing.	Bid details displayed for editing.	
BM_TC_003	Bid Accept	Verify the ability to accept a submitted bid.	1. Navigate to the list of submitted bids.	List of all bids should be visible	List of all bids is display	P
			2. Select a bid to accept. Click on the "Accept" button.	The selected bid should be accepted successfully.	Selected bid accepts successfully.	P
BM_TC_004	Bid Accept	Verify the ability to reject a submitted bid.	1. Navigate to the list of submitted bids.	List of all bids should be visible	List of all bids is display	P
			2. Select a bid to reject. Click on the "Reject" button.	The selected bid should be rejected successfully.	Selected bid rejects successfully.	P

BM_TC_005	Bid Status Update	Verify the bid status update after evaluation.	1. Navigate to the list of submitted bids.	List of all bids should be visible	List of all bids is display	P
			2. Verify the status of the evaluated bid.	The bid status should be updated to reflect the evaluation results.	Bid status updated to reflect the evaluation results.	P

5. Feedback Module

Test Case ID	Test Case Name	Test Case Description	Test Steps	Expected Result	Actual Result	Test Status (P/F)
FM_TC_001	Submit Feedback	Verify successful of submitted feedback	1. Navigate to the feedback page.	Feedback page should be display	Feedback page is display	P
			2. Fill in the required details for the feedback. Click on the "Submit" button.	Feedback should be submitted successfully	Show message about feedback submission	P
FM_TC_002	View all Feedbacks	Verify successful of view feedbacks.	1. Navigate to the Feedback List page.	List of Feedback should be display	List of feedback is display	P

5.4 Defect Report

Defect ID	Module	Test Case ID	Description	Severity	Priority	Status
D-001	Vendor	VM_TC_003	Error message not displayed for invalid registration details	Medium	High	Open
D-002	Vendor	VM_TC_004	Error message not displayed for invalid login credentials	Low	Medium	Open
D-003	Tender	TM_TC_003	Tender search functionality does not return results	High	High	In Progress
D-004	Bids	BM_TC_002	Editing a bid does not save changes	Medium	High	Open
D-005	Notice	NM_TC_001	Newly created notice is not displayed in the notice board	High	High	Open

6. Limitation of proposed system

- **Initial Investment:** Implementing the system may require a significant initial investment in terms of development costs, software licenses, and hardware infrastructure.
- **User Adoption:** Users may face challenges in adapting to the new system, especially if they are accustomed to manual or legacy processes. Adequate training and support may be needed to ensure smooth adoption.
- **Technical Dependencies:** The system's functionality may rely on external factors such as network connectivity, third-party APIs, or integration with other systems. Any issues with these dependencies could impact system performance and reliability.
- **Scalability:** While the system may be designed to scale efficiently, rapid growth in users or data volume could present scalability challenges. Regular monitoring and capacity planning may be required to address scalability issues proactively.
- **Security Risks:** Despite robust security measures, the system may still be vulnerable to security threats such as cyber-attacks, data breaches, or unauthorized access. Ongoing security assessments and updates are necessary to mitigate these risks.
- **Regulatory Compliance:** Ensuring compliance with relevant regulations and industry standards may pose challenges, especially if there are frequent changes or updates to compliance requirements.
- **Integration Challenges:** Integrating the system with existing enterprise systems or third-party applications may be challenging due to differences in data formats, protocols, or APIs. Compatibility issues and data migration complexities could arise during the integration process.

7. Proposed Enhancement

- **Mobile Compatibility:** Enhance the system to be fully compatible with mobile devices, allowing users to access and interact with the system from anywhere, at any time, using smartphones or tablets.
- **Advanced Reporting and Analytics:** Integrate advanced reporting and analytics capabilities, including predictive analytics and data visualization tools, to provide deeper insights into tender management processes and trends.
- **Artificial Intelligence (AI) Integration:** Incorporate AI technologies such as natural language processing (NLP) for automated document analysis, sentiment analysis for evaluating bidder responses, or machine learning algorithms for predicting tender outcomes.
- **Blockchain Integration:** Implement blockchain technology to enhance the security, transparency, and traceability of tender-related transactions, ensuring immutability and integrity throughout the procurement process.
- **Enhanced Collaboration Tools:** Introduce collaborative features such as real-time document editing, chat functionality, or virtual meeting rooms to facilitate seamless communication and collaboration between stakeholders involved in the tendering process.
- **Supplier Performance Management:** Develop modules for tracking and evaluating supplier performance based on metrics such as delivery timeliness, quality of goods or services, and compliance with contractual terms.
- **Workflow Automation:** Automate repetitive tasks and workflows within the system, such as document approval processes, bid scoring, or contract management, to improve efficiency and reduce manual effort.
- **Enhanced Security Measures:** Implement additional security measures such as multi-factor authentication, encryption of sensitive data at rest and in transit, and continuous monitoring for suspicious activities to strengthen the system's security posture.
- **User Experience Improvements:** Continuously enhance the user interface and user experience based on user feedback and usability testing, ensuring an intuitive and seamless experience for all system users.

8. Conclusion

In conclusion, the Tender Management System represents a significant advancement in modernizing procurement processes and streamlining tender management within organizations. Through its comprehensive features and functionalities, the system addresses the complexities and challenges associated with traditional tender management methods.

The system offers a centralized platform that enables efficient creation, publication, submission, and evaluation of tenders, fostering transparency, efficiency, and accountability throughout the tendering process. Stakeholders, including administrators, vendors, and bidders, benefit from enhanced communication channels, real-time access to tender information, and robust security measures to protect sensitive data.

Furthermore, the system's reporting and analytics capabilities provide valuable insights into tender activities, performance metrics, and trends, empowering organizations to make data-driven decisions and optimize procurement processes.

As organizations continue to embrace digital transformation and seek innovative solutions to streamline operations, the Tender Management System stands as a reliable tool for driving efficiency, reducing costs, and achieving better outcomes in procurement activities.

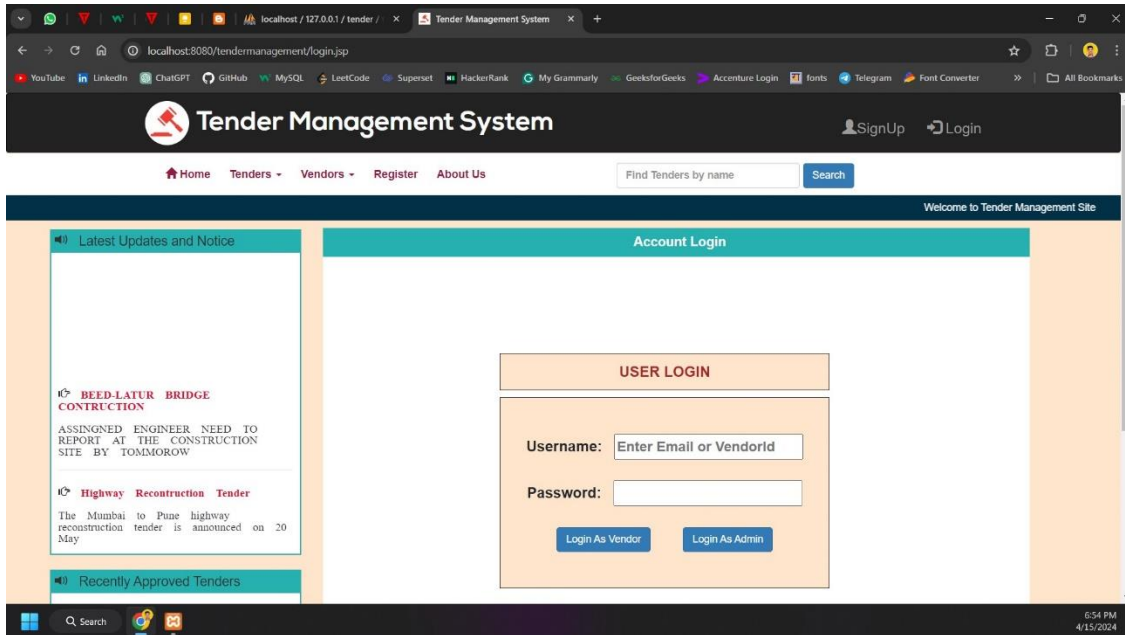
In conclusion, the Tender Management System offers a strategic advantage to organizations by revolutionizing the way tenders are managed, ensuring competitiveness, compliance, and success in the procurement landscape.

9. Bibliography

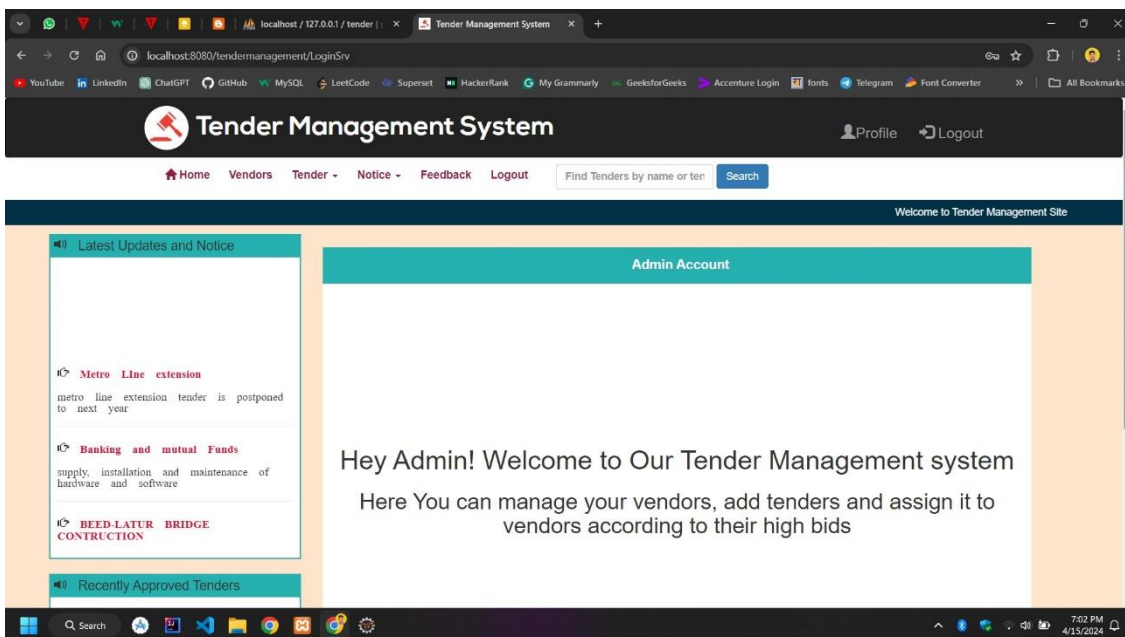
- www.Google.com
- www.youtube.com
- www.javatpoint.com
- www.stackoverflow.com
- www.w3schools.com
- <https://chat.openai.com/>

10. User Manual

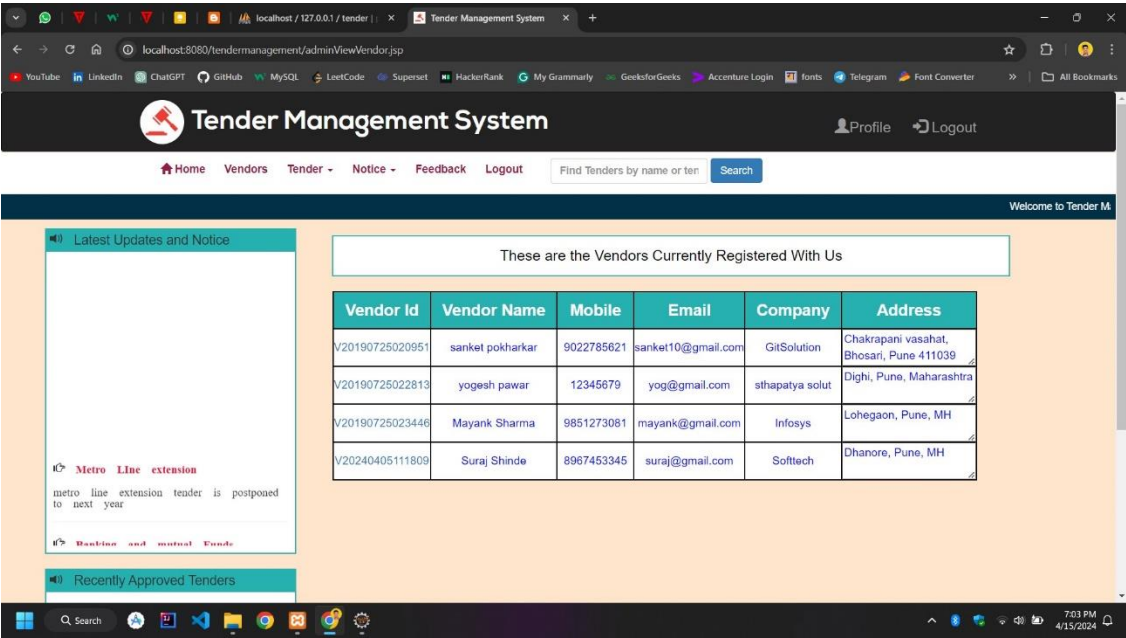
Admin Login: The admin login module provides secure access for administrators to enter a system or platform, typically requiring a username and password, granting privileged control over administrative functions and settings.



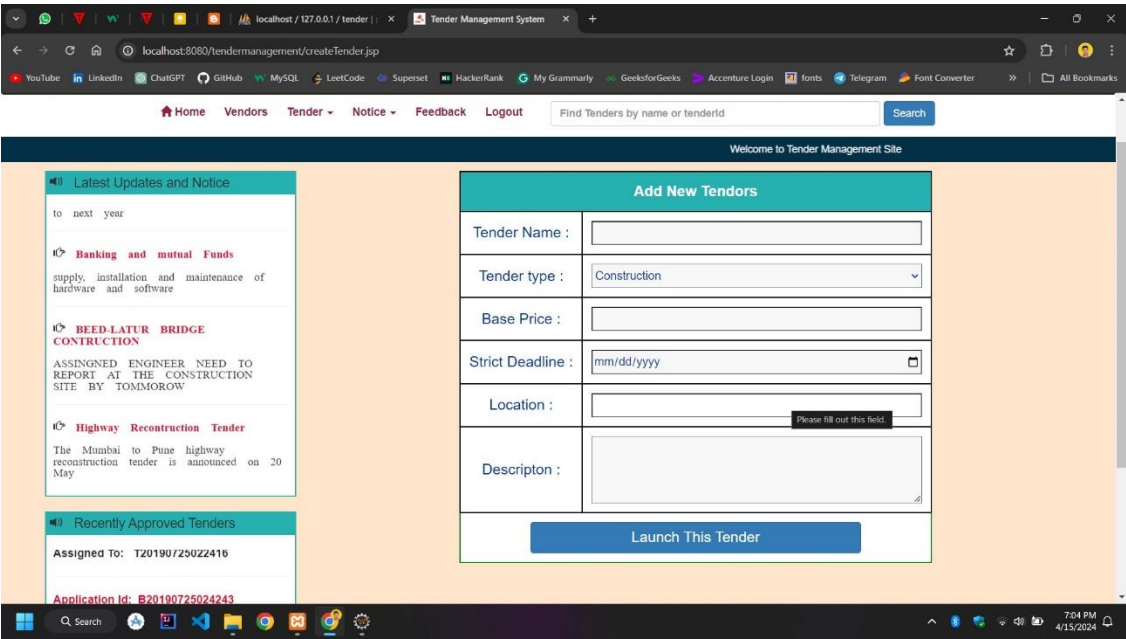
Admin Home: The admin home module serves as the dashboard or landing page for administrators upon logging in, presenting essential tools, data summaries, and navigation options tailored to manage administrative tasks efficiently.



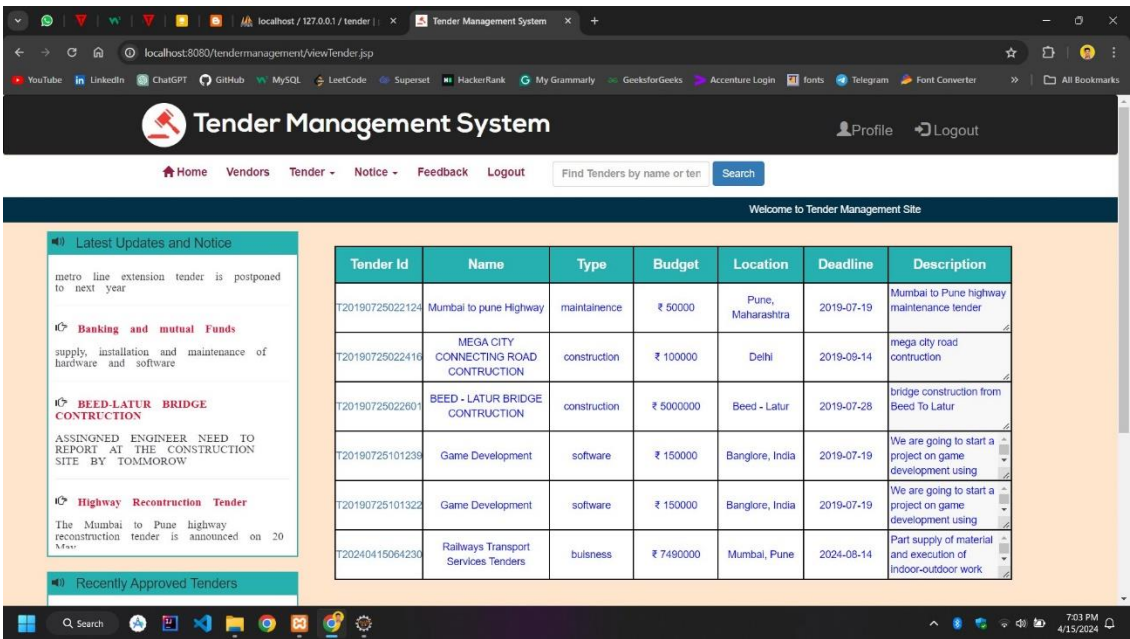
View Vendor: The Vendor List Page displays a comprehensive list of registered vendors within the Tender Management System. It provides essential vendor details such as names, contact information, and company.



Create Tender: The Create Tender Page allows administrators to efficiently generate new tender opportunities within the system. Administrators can input key details such as tender title, description, and deadlines through a user-friendly interface.



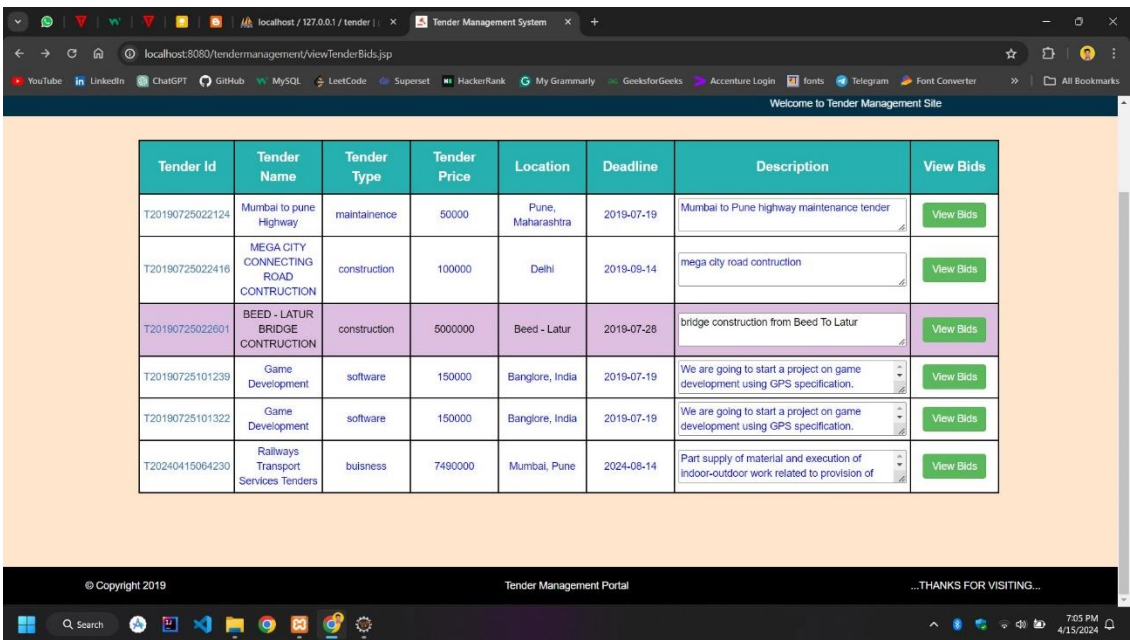
View Tenders: The Tender List Page provides a comprehensive overview of all active tenders within the system. Administrators can easily navigate through the list and access essential details such as tender titles, deadlines, and current statuses.



The screenshot shows the 'Tender Management System' interface. The main content area displays a table of active tenders. The table has columns for Tender Id, Name, Type, Budget, Location, Deadline, and Description. The data is as follows:

Tender Id	Name	Type	Budget	Location	Deadline	Description
T20190725022124	Mumbai to pune Highway	maintenance	₹ 50000	Pune, Maharashtra	2019-07-19	Mumbai to Pune highway maintenance tender
T20190725022416	MEGA CITY CONNECTING ROAD CONSTRUCTION	construction	₹ 100000	Delhi	2019-09-14	mega city road construction
T20190725022601	BEED - LATUR BRIDGE CONSTRUCTION	construction	₹ 5000000	Beed - Latur	2019-07-28	bridge construction from Beed To Latur
T20190725101239	Game Development	software	₹ 150000	Banglore, India	2019-07-19	We are going to start a project on game development using
T20190725101322	Game Development	software	₹ 150000	Banglore, India	2019-07-19	We are going to start a project on game development using
T20240415064230	Railways Transport Services Tenders	business	₹ 7490000	Mumbai, Pune	2024-08-14	Part supply of material and execution of indoor-outdoor work

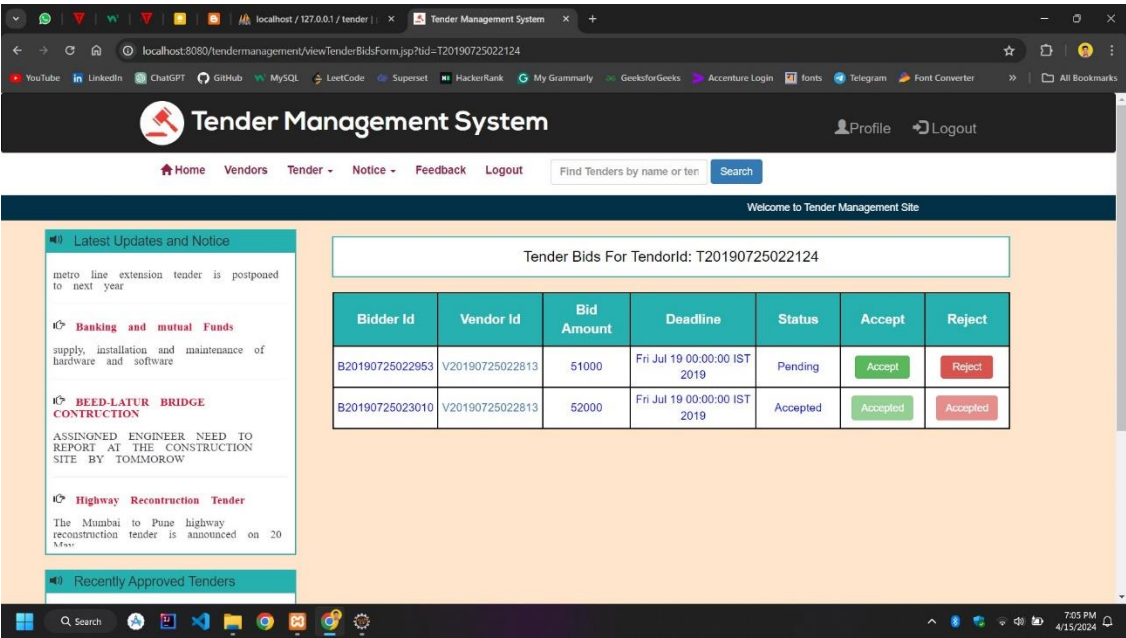
View Tender Bids: The "View Tender Bids" page offers administrators a detailed overview of bids submitted for a specific tender. It provides key information such as bid amounts, proposal details, and bidder identities.



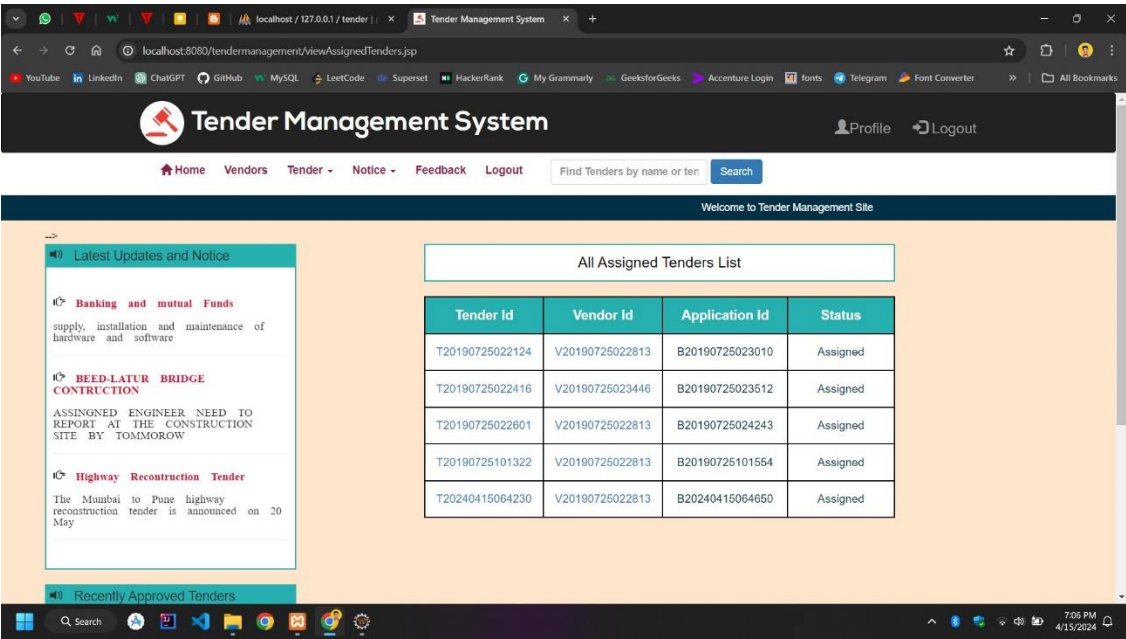
The screenshot shows the 'View Tender Bids' page. The main content area displays a table of bids for various tenders. The table has columns for Tender Id, Tender Name, Tender Type, Tender Price, Location, Deadline, Description, and a 'View Bids' button. The data is as follows:

Tender Id	Tender Name	Tender Type	Tender Price	Location	Deadline	Description	View Bids
T20190725022124	Mumbai to pune Highway	maintenance	50000	Pune, Maharashtra	2019-07-19	Mumbai to Pune highway maintenance tender	View Bids
T20190725022416	MEGA CITY CONNECTING ROAD CONSTRUCTION	construction	100000	Delhi	2019-09-14	mega city road construction	View Bids
T20190725022601	BEED - LATUR BRIDGE CONSTRUCTION	construction	5000000	Beed - Latur	2019-07-28	bridge construction from Beed To Latur	View Bids
T20190725101239	Game Development	software	150000	Banglore, India	2019-07-19	We are going to start a project on game development using GPS specification.	View Bids
T20190725101322	Game Development	software	150000	Banglore, India	2019-07-19	We are going to start a project on game development using GPS specification.	View Bids
T20240415064230	Railways Transport Services Tenders	business	7490000	Mumbai, Pune	2024-08-14	Part supply of material and execution of indoor-outdoor work related to provision of	View Bids

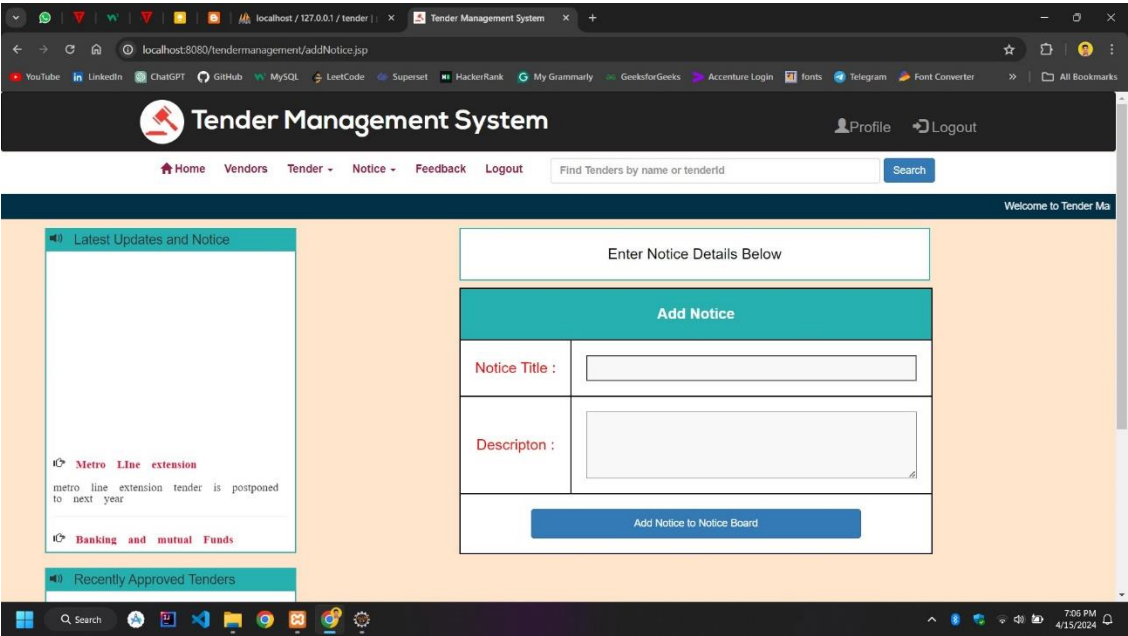
Accept Bids Page: The "Accept Bids" page empowers administrators to review and accept bids submitted for a tender. Administrators can carefully evaluate bid details and select winning bids based on predefined criteria, facilitating efficient tender awarding.



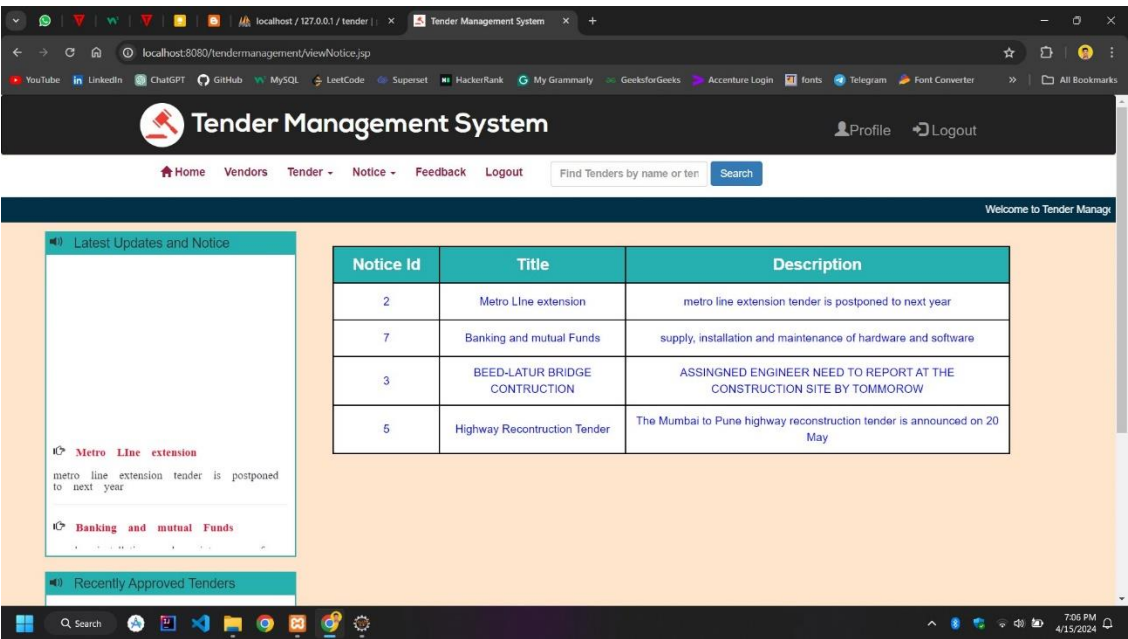
Assigned Tenders: The "Assigned Tenders" page provides administrators with a consolidated view of tenders assigned to specific vendors. It displays essential details such as vendor id, tender id, and assignment statuses.



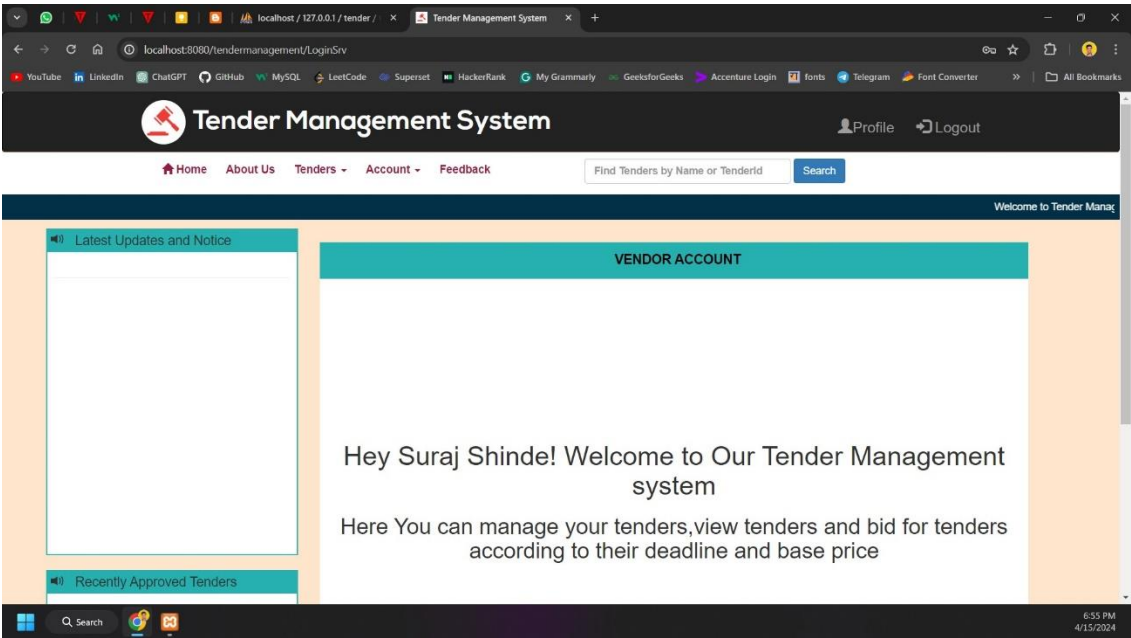
Create Notice: The "Create Notice" page enables administrators to efficiently compose and publish important announcements within the system. Administrators can input notice details such as title, content, and visibility settings through a user-friendly interface.



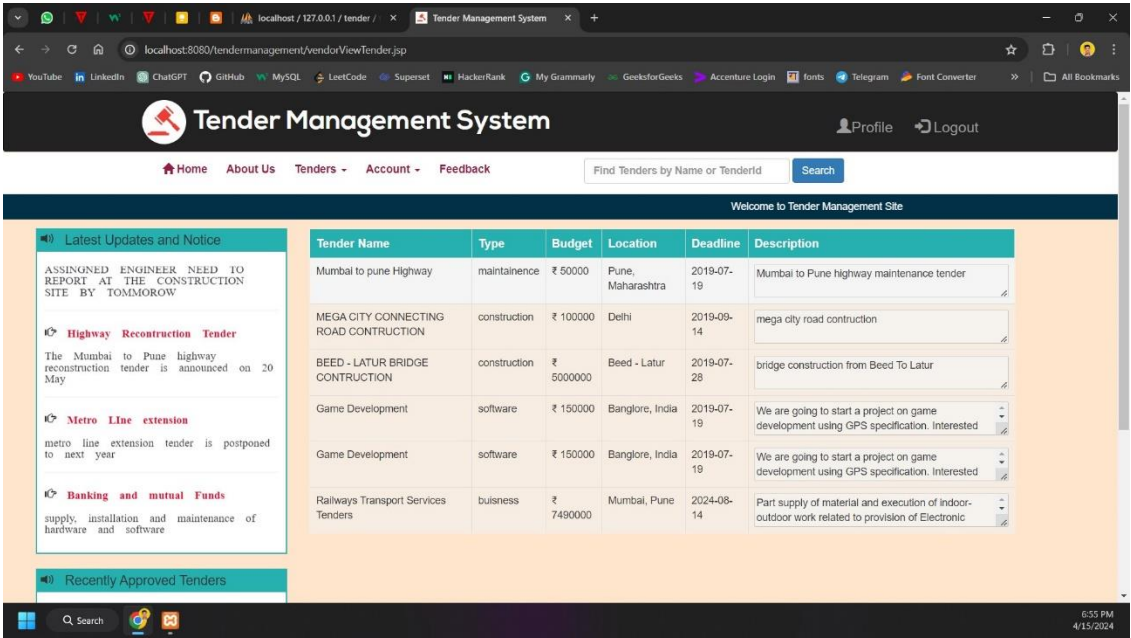
View Notice: The "View Notice" page provides users with easy access to important announcements and updates within the system. Users can conveniently browse through a list of notices, view their details.



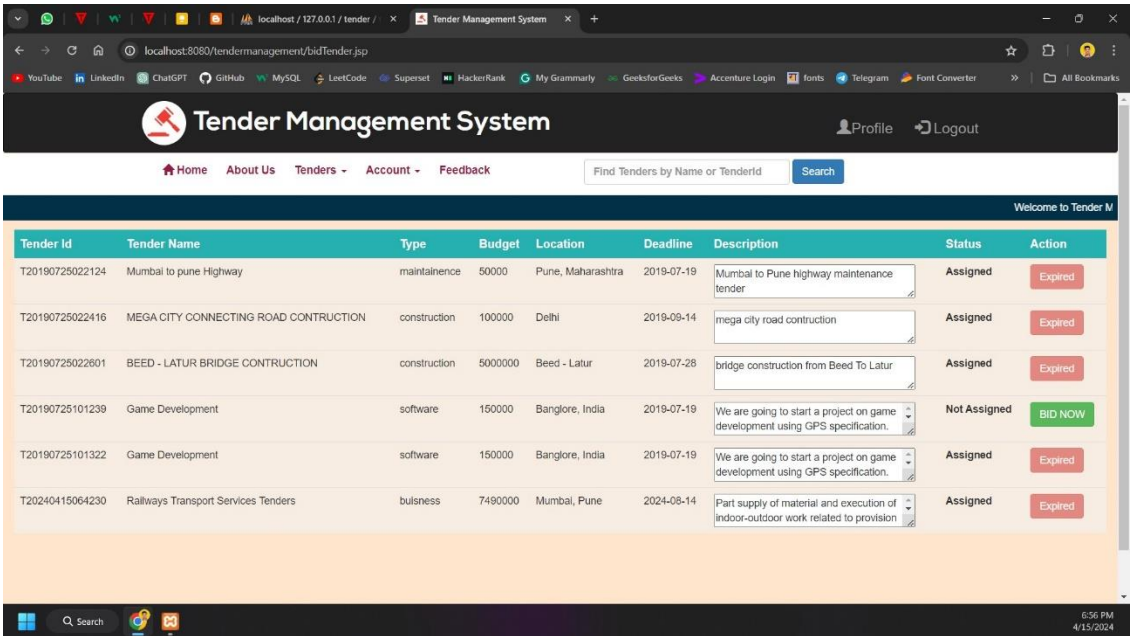
Vendor Home: The "Vendor Home" page serves as a centralized hub for vendors within the Tender Management System. It offers vendors quick access to essential features such as tender listings, bid submission, and profile management.



View Tenders: The Tender List Page provides a comprehensive overview of all active tenders within the system. Administrators can easily navigate through the list and access essential details such as tender titles, deadlines, and current statuses.

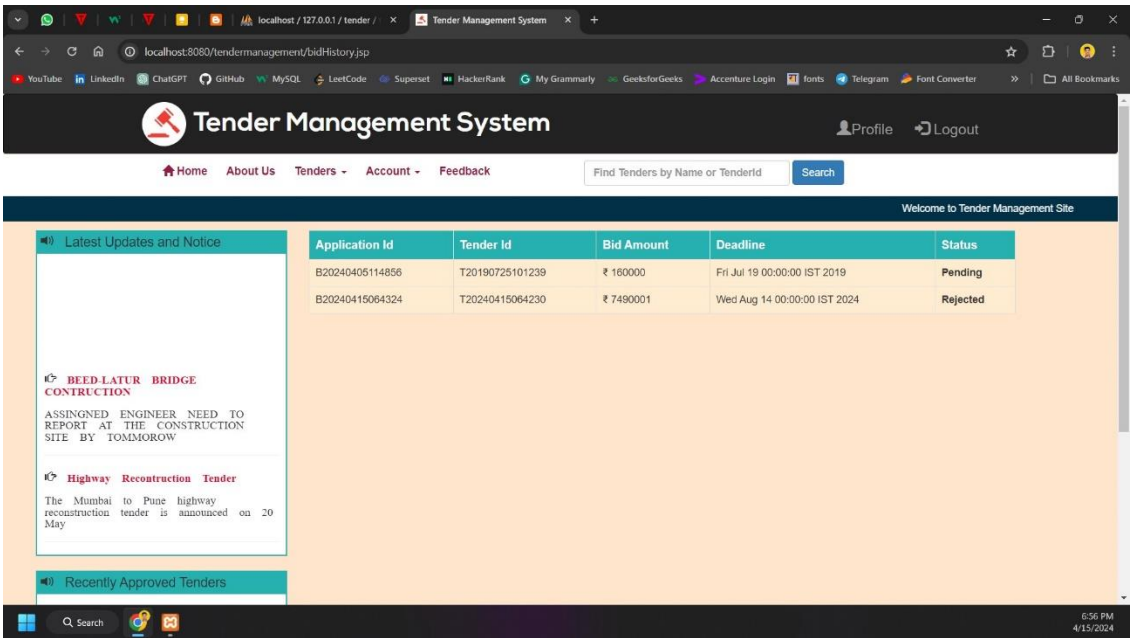


Bid for Tender: The "Bid for Tender" page enables vendors to submit competitive bids for available tender opportunities. Vendors can input bid details such as pricing, proposal, and additional information through a straightforward interface.



Tender Id	Tender Name	Type	Budget	Location	Deadline	Description	Status	Action
T20190725022124	Mumbai to pune Highway	maintenance	50000	Pune, Maharashtra	2019-07-19	Mumbai to Pune highway maintenance tender	Assigned	Expired
T20190725022416	MEGA CITY CONNECTING ROAD CONSTRUCTION	construction	100000	Delhi	2019-09-14	mega city road construction	Assigned	Expired
T20190725022601	BEED - LATUR BRIDGE CONSTRUCTION	construction	5000000	Beed - Latur	2019-07-28	bridge construction from Beed To Latur	Assigned	Expired
T20190725101239	Game Development	software	150000	Banglore, India	2019-07-19	We are going to start a project on game development using GPS specification.	Not Assigned	BID NOW
T20190725101322	Game Development	software	150000	Banglore, India	2019-07-19	We are going to start a project on game development using GPS specification.	Assigned	Expired
T20240415064230	Railways Transport Services Tenders	business	7490000	Mumbai, Pune	2024-08-14	Part supply of material and execution of indoor-outdoor work related to provision	Assigned	Expired

Bidding History: The "Bidding History" page provides vendors with a comprehensive overview of their past bidding activities within the system. Vendors can easily track and review details such as bid amounts, tender titles, and bid statuses.



Application Id	Tender Id	Bid Amount	Deadline	Status
B20240405114856	T20190725101239	₹ 160000	Fri Jul 19 00:00:00 IST 2019	Pending
B20240415064324	T20240415064230	₹ 7490001	Wed Aug 14 00:00:00 IST 2024	Rejected

View Profile: The "View Profile" page offers users a comprehensive overview of their personal and professional information within the system. Users can access and review details such as contact information, company names, and account preferences.

The screenshot displays the 'View Profile' page of the Tender Management System. The page is divided into three main sections. On the left, there are two sections: 'Latest Updates and Notice' and 'Recently Approved Tenders'. The 'Latest Updates and Notice' section contains three items: 'Metro Line extension' (postponed to next year), 'Banking and mutual Funds' (supply, installation and maintenance of hardware and software), and 'BEED-LATUR BRIDGE CONSTRUCTION' (assigned engineer need to report at the construction site by tomorrow). The 'Recently Approved Tenders' section shows two items: 'Application Id: B20240415064650' assigned to 'T20240415064230' and 'Application Id: B20190725023512' assigned to 'T20190725022416'. The right section is titled 'User Profile' and contains a table with the following details: Vendor Id: V20240405111809, Vendor Name: Suraj Shinde, Mobile No.: 8967453345, Email Id: suraj@gmail.com, Address: Dhanore, Pune, MH, and Company Name: Softech. Below the table is a link 'Click Here To Update Profile'.

User Profile	
Vendor Id :	V20240405111809
Vendor Name:	Suraj Shinde
Mobile No. :	8967453345
Email Id :	suraj@gmail.com
Address :	Dhanore, Pune, MH
Company Name :	Softech
Click Here To Update Profile	

Update Profile: The "Update Profile" page empowers users to conveniently modify and maintain their personal and professional information within the system. Users can easily update details such as contact information, company name, and account preferences through an intuitive interface

The screenshot displays the 'Update Profile' page of the Tender Management System. The page is divided into three main sections. On the left, there are two sections: 'Latest Updates and Notice' and 'Recently Approved Tenders'. The 'Latest Updates and Notice' section contains three items: 'Metro Line extension' (postponed to next year), 'Banking and mutual Funds' (supply, installation and maintenance of hardware and software), and 'BEED-LATUR BRIDGE CONSTRUCTION' (assigned engineer need to report at the construction site by tomorrow). The 'Recently Approved Tenders' section shows two items: 'Application Id: B20190725023512' assigned to 'T20190725022416' and 'Application Id: B20190725024243' assigned to 'T20190725022416'. The right section is titled 'Update Profile' and contains a form to edit user details. The form has a header 'Edit Details And Click on Update' and a table with the following details: Vendor Id: V20240405111809, Vendor Name: Suraj Shinde, Mobile No.: 8967453345, Email Id: suraj@gmail.com, Address: Dhanore, Pune, MH, and Company Name: Softech. Below the table is a button 'Update Profile'.

Update Profile	
Vendor Id :	V20240405111809
Vendor Name:	Suraj Shinde
Mobile No. :	8967453345
Email Id :	suraj@gmail.com
Address :	Dhanore, Pune, MH
Company Name :	Softech
Verify Password:	
Update Profile	